Data Analysis Methods for Copy Number Discovery and Interpretation

Tomas William Fitzgerald
Wellcome Trust Sanger Institute
Cranfield University

This dissertation is submitted for the degree of
Doctor of Philosophy
October 2014

Declaration

I hereby declare that this dissertation describes my own work and does not include work that has been done in collaboration, except where explicitly stated in the text. The thesis contains material that has not been submitted for a degree or diploma or any other qualification at any other university.

Tomas William Fitzgerald
23 October 2014

Abstract

Copy number variation (CNV) is an important type of genetic variation that can give rise to a wide variety of phenotypic traits. Differences in copy number are thought to play major roles in processes that involve dosage sensitive genes, providing beneficial, deleterious or neutral modifications to individual phenotypes. Copy number analysis has long been a standard in clinical cytogenetic laboratories. Gene deletions and duplications can often be linked with genetic Syndromes such as: the 7q11.23 deletion of Williams-Bueren Syndrome, the 22q11 deletion of DiGeorge syndrome and the 17q11.2 duplication of Potocki-Lupski syndrome. Interestingly, copy number based genomic disorders often display reciprocal deletion / duplication syndromes, with the latter frequently exhibiting milder symptoms. Moreover, the study of chromosomal imbalances plays a key role in cancer research.

The datasets used for the development of analysis methods during this project are generated as part of the cutting-edge translational project, Deciphering Developmental Disorders (DDD). This project, the DDD, is the first of its kind and will directly apply state of the art technologies, in the form of ultra-high resolution microarray and next generation sequencing (NGS), to real-time genetic clinical practice. It is collaboration between the Wellcome Trust Sanger Institute (WTSI) and the National Health Service (NHS) involving the 24 regional genetic services across the UK and Ireland. Although the application of DNA microarrays for the detection of CNVs is well established, individual change point detection algorithms often display variable performances. The definition of an optimal set of parameters for achieving a certain level of performance is rarely straightforward, especially where data qualities vary.

A combined change point detection package, CNsolidate, is developed as part of this research, which makes use of multiple weighted algorithms. Using this approach it is possible to rank detections based on differential weighting functions between component algorithms, which substantially improves the Type I and Type II error rates relative to other approaches. The DDD project has a responsibility to ensure accuracy and consistency in all data sets reported back to the UK clinical teams. A novel method is developed to allow the accurate tracking of array-CGH data generated as part of the DDD project using copy number tagging SNPs. Additionally, the DDD makes use of a number of advanced variant prediction approaches, including the accurate assignment of population based frequency estimates and the ranking of variants based on their relevance to health. In addition to microarray data on approximately 1,000 control individuals generated specifically for the DDD project, a consensus approach is used to generate common CNV reference sets, calculating frequency estimates across studies displaying differential sensitivities across the genomic range. Finally a rule-based approach for ranking CNVs in terms of their potential clinical significance is implemented to facilitate the feedback of clinically relevant CNVs to patients via the NHS regional genetic services.

Acknowledgments

There are a large number of inspiring individuals I would like to thank, who have provided extensive support and guidance during the development of all these methods.

Current Supervisors

Matthew E Hurles[1] & Fady Mohareb[2].

Ex Supervisors

Nigel P Carter[1] & Lee Larcombe[2] & Conrad Bessant[2].

Other Sources of Inspiration

Jeffery Barrett[1], Helen Firth[1], Caroline Wright[1], Kate Morley[1], Eugene Bragin[1],Parthiban Vijayarangakannan[1] & Margriet van Kogelenberg[1].

General Acknowledgments

DDD Informatics & High-throughput array-CGH Teams[1]

Dedication

Most of all I would like to thank my family who provide me with endless support and happiness with every new day.

I would like to dedicate this work to my father, William John Fitzgerald, who remains my strongest inspiration in most areas, including data analysis problems.

To my father, I thank you for your constant and patient guidance through life.

On the most westerly Blasket
In a dry-stone hut
He got this air out of the night.

Strange noises were heard
By others who followed, bits of a tune
Coming in on load weather

Though nothing like melody.
He blamed their fingers and ear
As unpractised, their fiddling easy

For he had gone alone into the island
And brought back the whole thing.
The house throbbed like his full violin.

So whether he calls it sprint music
Or not, I don't care. He took it
Out of wind off mid-Atlantic.

Still he maintains, from nowhere.
It comes off the bow gravely,
Rephrases itself into the air.

The Given Note by Seamus Heaney

Table of Contents

Nomenclature

Abbreviations

BAC	Bacterial artificial chromosome
BAF	Biallele frequency
CGH	Comparative genomic hybridisation
CNA	Copy number alteration
CNV	Copy number variation
CNVE	Copy number variable event
CNVR	Copy number variable region
CUSUM	Cumulative sum control chart
DDD	The Deciphering Developmental Disorders Project
DDG2P	DD gene to phenotype database
DGV	Database of genomic variation
dLRs	Derivative log ratio spread
DOP-PCR	Degenerate-oligonucleotide-primed PCR
DP	Dynamic programming algorithm
EM	Expectation maximization
FDR	False discovery rate
FISH	Fluorescence in-situ hybridisation
FNR	False negative rate
FosTes	Fork stalling and template switching
FPR	Mean false positive rate
GEL	Genomics England – The 100,000 Genomes Project
GRC	The Genome Reference Consortium
GWAS	Genome wide association studies
HMM	Hidden markov model
HPO	Human phenotype ontology
ISCA	International Standards for Cytogenomic Arrays Consortium
L2R	Log2 Ratio
LCR	Low copy repeats
LD	Linkage disequilibrium
LIMS	Laboratory information management system
LOH	loss of heterozygosity
LRR	Log R Ratio
LSF	Load sharing facility
MAD	Median absolute deviation
MAF	Minor allele frequency
MAP	Segmental maximum a posteriori
MMBIR	Microhomology mediated break induced repair
NAHR	Non-allelic-homologous recombination
NGS	Next generation sequencing
NHEJ	Non-homologous end joining
NHS	The National Health Service
POLYPHEN	Polymorphism phenotyping
QC	Quality control
ROC	Receiver operator curve

SIFT	Sorting intolerant from tolerant
SNP	Single nucleotide polymorphism
SQL	Structured query language
TE	Transposable element
TPR	True positive rate
UK10K	The UK10K Project
UKBS	The UK Blood Donor Service
UPD	Segmental uniparental disomy
VOUS	Variants of uncertain significance
WTCCC	The Wellcome Trust Case Control Consortium
WTSI	The Wellcome Trust Sanger Institute

List of Figures

List of Tables

List of Equations

1 | Introduction and Literature Review

1.1 INTRODUCTION

Comparative Genomic Hybridisation (CGH) was developed around the early 1990s and has been applied for over a decade to screen for chromosomal aberrations in tumour samples [1]. Originally, this type of CGH would have been carried out using metaphase chromosomes spread on glass slides and co-hybridising labelled DNA from both test and reference samples. This allowed for a direct comparison between the two samples using a confocal microscope to check for large-scale rearrangements of genetic material within each sample. By using specific probes, complementary to important sequences of DNA (e.g. the b52 gene), labelled with one of a variety of fluorophores it became possible to check for both gene deletion and amplification in a range of different samples. However, the functional resolution remained relatively low (5-15Mb) and both the experimental processing and data interpretation were challenging, requiring a high degree of skill and expertise [2]. As the technology progressed, and new resources from large scale sequencing projects became available (BAC clones), in the mid 1990s, the field of array-CGH (first called matrix-CGH) was born. This methodology was first described in 1997 where the production of the first CGH arrays and protocols used during this new method, matrix-CGH, were outlined [3, 4]. However, these initial methods for microarray construction were time consuming and difficult to perform. Later new methods were developed to increase the speed and ease of microarray construction [5].

With the emergence of genomic resources generated as a result of the International Human Genome Sequencing project, the flexibility of microarray design was dramatically improved. It was now possible, for the first time, to use large insert clones, such as bacterial artificial chromosomes (BACs), to fully span the entire human genome using overlapping (tiling) clones. At the same time the improvement of microarray spotting technologies allowed for the metaphase chromosome spreads to be replaced by glass slides spotted with DNA fragments mapped to precise locations along the genome and arrayed on the slide in a grid formation. This approach initially increased the resolution of CGH methods by more than tenfold and allowed for smaller genetic differences to be assessed between two samples. The resolution of CGH was improved from approximately a maximum of 5Mb when using metaphase chromosome spreads to approximately 170Kb with BACs and 40Kb with fosmid based insert clone arrays. Furthermore, it was now possible to imagine that two samples could be compared on one microarray to screen for genetic changes across the entire genome. Indeed, this approach was embraced by the community and has been used to screen for micro-duplications and micro-deletions in patients with constitutional rearrangements [6, 7], cross-species studies of evolution [8-10], and studies of migration and ethnic evolution in humans [11].

The resources used for the production of the first microarrays (large insert clones) were made widely available and have allowed for new microarrays

covering the entire genome to be produced in increasingly high resolutions [12]. On top of the increase in resolution, the ease of experimental and laboratory processing was improved and the number of experiments possible to run every day was significantly increased. As this new emerging technology grew, its use became more wide spread and large-scale studies of copy number variation (CNV) were coined. The increase in both resolution and experimental processing ease had meant that CGH changed from a relatively time consuming, specialist field to a high-throughout method suitable for use in large-scale studies. However, the technology took a relatively long time to mature and even though it was utilized during small-scale studies from as early as the mid 1990s it was not until 2006 that the first comprehensive map of CNV in the human genome was published using a whole genome tiling BAC array [13].

Recent advancements in array-based comparative genomic hybridisation technology enable entire genomes to be scanned for copy number changes using high-resolution oligonucleotide tiling arrays or single nucleotide polymorphism (SNP) genotyping arrays. In particular, it is now possible to design the content of arrays with a high degree of flexibility as a number of companies offer rapid custom array generation as a standard service. Due to the particular needs of specific experimental questions, microarrays frequently contain oligonucleotide probes displaying variable performance particularly when designs are targeted towards regions of the genome with complex architecture. For example, regions containing repetitive sequences are important for studies of both copy number variation (CNV) and copy number alterations (CNA) and often need to be included in array designs. Replication hot spots contain large numbers of repetitive structures and often underlie the mechanisms for copy number break point formation, which include non-allelic-homologous recombination (NAHR), non-homologous end joining (NHEJ) and fork stalling and template switching (FosTes) mechanisms [14].

Studies on CNV are often used to search for rare recurrent rearrangements associated with rare disease [15]. Most of the studies to date have been relatively small scale due, in part, to the relative rarity of the disorders under study. More recently with resources such as the DECIPHER database becoming available [16], facilitating the sharing of genetic findings and international collaboration, an increased potential to study rare disease on a large-scale has been realised. SNP-genotyping arrays have been used extensively in large scale genome wide association studies (GWAS) to identify common genetic factors influencing health and disease. The GWAS study is most often used to identify susceptibility loci for common diseases, including Crohn's disease [17], type1 [18] and type2 [19] diabetes along with a large variety of others. It is possible to detect CNV using genome-wide SNP arrays however, the detection power is often limited and the development of new, high-performance detection algorithms is an active area of research [20]. Currently the only reliable array-based technologies available for the detection of segmental uniparental disomy (UPD) or copy number loss of heterozygosity (LOH) are SNP genotyping chips [21, 22]. Segmental UPD occurs when a person receives two copies of a chromosome, or a chromosomal region, from one parent and no copies from the other parent. This mode of genetic transfer is known to cause several genomic disorders, including

Prader-Willi, Angelman and Beckwith Wiedemann syndromes [23]. Loss of heterozygosity (LOH) is most often observed in conjunction with UPD, however, a subtle, yet often overlooked fundamental difference is that LOH can occur via random mutation events. Although the potential to extend genome-wide association studies to include CNV has been realised for a relatively long period of time [24], its large-scale application has been significantly hampered by a number of factors. Some of these factors include the poor understanding of 'benign' (neutral) CNVs, the heterogeneity of reference data, the differences in sensitivity across platforms as well as the relative difficulty of experimental handling, data analysis and data interpretation [25-28]. As a result of currently active large-scale studies on genomic variation, including the Deciphering Developmental Disorders (DDD) [29] and the International Standards for Cytogenomic Arrays Consortium (ISCA) [30] projects, the amount of genome wide copy number data available for data mining will increase dramatically in the coming years. Scientific research groups across the globe are currently developing methodologies to enable the utilisation of these new rich datasets [31-33]. It is highly likely that there will be marked improvements in both the number of available data analysis methods and in the ability to interpret these data in the near future.

DNA sequencing methods have been around for a long time but with the emergence of the next-generation sequencing (NGS) technologies came a massive reduction in the cost and speed of sequencing an entire genome [34, 35]. The number of high profile, large-scale studies using NGS to fine map genetic variation has seen huge increases in the last few years. Results from one of the most well known of these studies, the 1000 Genome Project, were published relatively recently [36]. There have also been a large number of publications describing both the use of NGS to identify new candidate gene loci and in the development of new sequence analysis methodologies [37-42]. The most commonly observed approach to using these large sequence datasets is to filter the detected variants against variation databases such as "DBSNP" [43], apply a predictive analysis on the functional effect of the remaining variants, using methods such as "Polymorphism Phenotyping" (POLYPHEN) and "Sorting Intolerant From Tolerant" (SIFT) [44], and then to search for a common variant between cases which display a similar phenotypic trait. Although this method has been shown to be very effective and has resulted in a large number of publications on new candidate gene loci [45], it is clear that there is a lot of potential information contained in the variants removed during these filtering processes that is currently being poorly utilised.

To fully understand the role that variation plays in making individuals different from one another, it is clearly essential that the necessary tools and expertise are available to allow the accurate interpretation of all the different flavours of genetic variation that can occur within an individual genome.

1.2 COPY NUMBER TECHNOLOGIES

1.2.1 Fluorescence in situ hybridisation

Fluorescence in-situ hybridisation (FISH) has been, and remains, an important tool for both clinical diagnostics and scientific research [46]. FISH uses fluorescently labelled probes to bind to complementary sequences of the genome and allows the visualisation of whole chromosomes using a confocal microscope. FISH can be used for a variety of purposes; it can detect large-scale genomic rearrangements, such as deletions, duplications, inversions, and translocations (see **Figure 1-1**). It was extensively used during the completion of the human genome sequence to provide ”anchors” to which the sequence was assembled around [47]. It remains the only reliable way to visualise genomic architecture in the context of whole chromosomes.

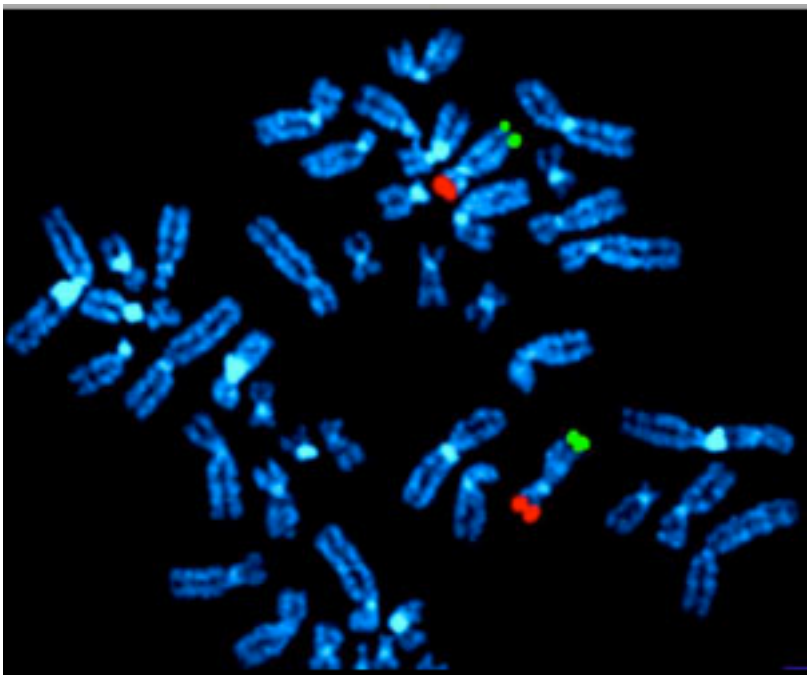


Figure 1-1: Fluorescence in situ hybridization (FISH) of metaphase human chromosomes. The probes are to the 5p telomere (red) and to the HAPLN1 gene at 5q14 (green) [cg.uchospitals.edu - Stephanie Mewborn].

1.2.2 Array Based Technologies

Array-CGH:

Nowadays, the use of clone-based array-CGH is less common and has, for the most part, been superseded by the higher resolution oligo nucleotide arrays. With an increase in the technology for microarray synthesis came the ability to produce in situ synthesised (oligo-nucleotide) microarrays [48]. The term oligo-nucleotide array refers to the type of manufacturing technique used. For oligo-nucleotide arrays the probes are short sequences of synthesised nucleotides designed to be complementary to specific regions of the genome. The

manufacture of oligo-nucleotide microarrays involves producing short oligo-nucleotide sequences by synthesising the sequence directly on the array surface (in situ). Four main manufacturers, Affymetrix, Illumina, Agilent and NimbleGen, produce a variety of different oligo-nucleotide microarrays tailored towards particular scientific questions.

Among the different types of commonly used arrays are SNP genotyping arrays, Gene Expression arrays, Comparative Genomic Hybridisation arrays (array-CGH), ChiP-on-chip arrays and alternative RNA splicing arrays. The design of oligo-nucleotide probes for microarrays was described as early as 2003 with the development of the GenomePRIDE software [49]. However, nowadays most of the array manufacturers provide there own software for custom microarray design, for example the e-array website provided by Agilent allows the rapid design and ordering of custom microarrays. These different arrays come in a variety of different formats and can be customised with a large amount of flexibility. The number of probes available for use can vary between the different manufacturers but offer a vast increase in the effective resolution compared to large insert clone based arrays. The highest resolution study and most comprehensive variation map to date was run on a set of 20 NimbleGen arrays comprising of 2.1 million probes, giving a total of 42 million features evenly spaced across the entire genome [50]. This array had a median probe spacing of 50bp and delivered an effective resolution of 250bp (5 probes). Compared to the maximum resolution of large insert clone arrays using fosmid clones (approx. 40kb) this offered a 160-fold increase in resolution. Compared to the original CGH methods, using metaphase chromosome spreads; the improvement in technology offered by oligo-nucleotide arrays has given a minimum of a 20,000 fold increase in genomic resolution.

Originally, the construction of spotted, large insert clone arrays relied on the extraction of DNA fragments from large numbers of bacterial cultures [5]. This process often resulted in only small amounts of DNA being available for spotting and although sufficient for small-scale projects was hardly scalable to the levels needed to produce whole genome tiling arrays or for the production of the numbers of arrays needed in large-scale studies. To address this issue a number of different methods were developed including the use of degenerate-oligonucleotide-primed PCR (DOP-PCR), developed at the Wellcome Trust Sanger Institute (WTSI) [51]. DOP-PCR was originally designed for general DNA amplification and as such allowed for the amplification of entire genomes in a single reaction. This meant it was possible to amplify large quantities of tens of thousands of DNA insert clones. After amplification, a common approach would be to spot the insert clones onto a glass slide utilising a robotic arm and an array of fine pins. The resulting grid formation on the slide contains a specific insert clone at each co-ordinate (see **Figure 1-2**). These spotted arrays are normally generated using a random distribution of insert clones across the array to ensure that any spatial effects are evenly distributed throughout the genome and not confined to, for example, a single chromosome [52]. The precise pattern of the grid layout and exact position of every insert clone is retained allowing for each individual clone to be mapped back to the genome during analysis. The

maximum resolution of spotted arrays is limited to approximately 40,000 spots due to the minimum size of the fine needle arrays used during robotic spotting.

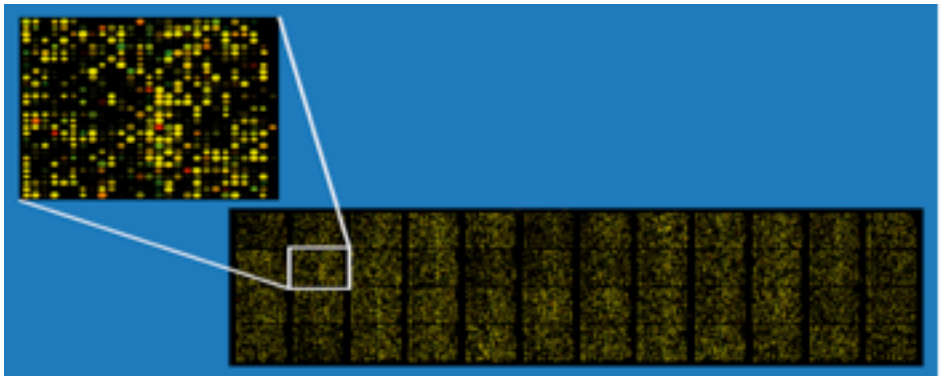


Figure 1-2 An example of a spotted microarray containing large insert clones.

For in situ oligo-nucleotide array fabrication there are a number of different approaches taken. The solid supporting surfaces can vary, with Agilent and NimbleGen using glass slides, Affymetrix using silicon chips (gene chips) and Illumnia opting for microscopic beads in place of a solid surface. The performance of the different platforms can vary and the choice of which to use would generally depend on the precise scientific question being addressed [53]. Agilent uses inkjet technology during the printing process, in which treated glass slides are used as the solid surface and oligo-nucleotide monomers are printed in extremely small volumes (picolitres) to precise co-ordinates on the slide (see **Figure 1-3**). The in situ synthesis then begins and the process ends once 60-mer sequences have been produced. These technologies have enabled Agilent to rapidly and accurately produce large numbers of microarrays and have put them among the world leaders for the production of microarrays. It is also possible to perform a photolithographic process without the use of a mask, for example NimbleGen use maskless lithography for oligo-nucleotide array fabrication. NimbleGen make use of a laser to shine light on specific co-ordinates of the solid surface with a high degree of precision.

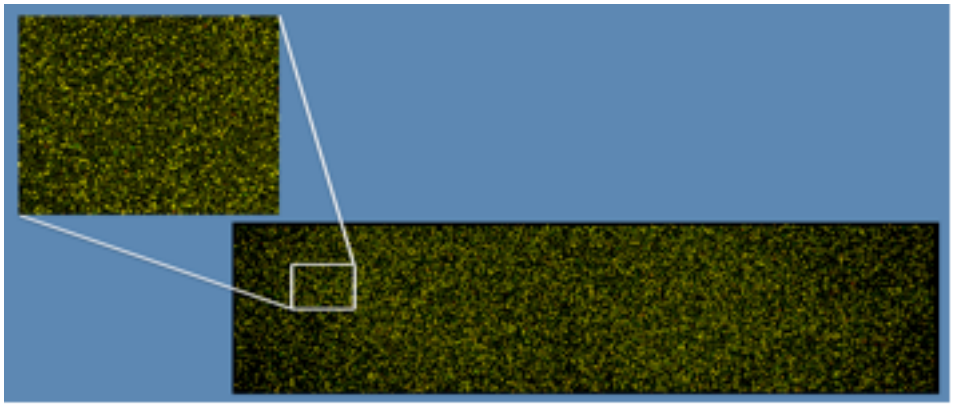


Figure 1-3 An example a printed oligo-nucleotide microarray fabricated using the Agilent SurePrint method.

The technique of array-CGH involves, by definition, the comparison of DNA samples on an array. Irrespective of the type of array being used the same basic methodology applies. Both a test and reference DNA are differentially labelled with one of a number of different flurophores (usually either Cy3 or Cy5). This process is most often performed, but not necessarily, by an overnight enzymatic reaction to allow the incorporation of labelled di-deoxy nucleotide molecules into both the test and reference DNA in equal quantities. The removal of un-incorporated nucleotides can then be achieved via filtration using a specific type of filtration column (Microcon YM-30 Millipore). The labelled DNA samples would then proceed through both a precipitation and resuspension procedure to result in both samples being mixed together in the same tube in the presence of some hybridisation buffer. The mixture can then be denatured, applied to the microarray (see **Figure 1-4**) and incubated over a period time at a particular temperature to allow hybridisation of the DNA molecules to the array [54].

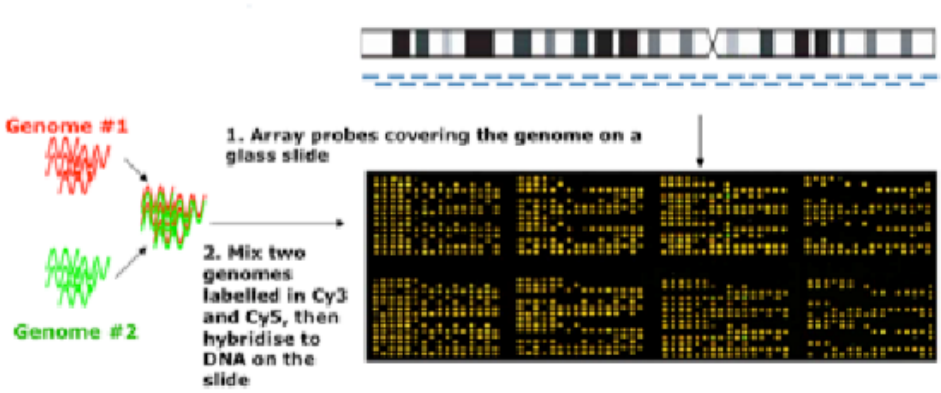


Figure 1-4 Illustration of applying fluorescently labelled DNA to a glass slide.

After hybridisation has completed and the array has been removed from incubation it needs to go through various washing steps to remove any un-specifically bound DNA fragments. These washes generally include both a stringent and non-stringent wash buffer and agitation of the microarray within each solution for a specific period of time. After the washing steps, the arrays need to be carefully dried and scanned by a high-resolution laser powered scanner (normally Agilent or Axon scanners). The scanning of the microarray results in a high-resolution image containing a spot for each probe on the array with a specific intensity and colour. This image can then be analysed using various different image analysis software to produce two intensity values for each spot (one for each flurophore used). The relative intensity of each dye for each spot on the array can then be converted to a log2 ratio and used to determine the relative amount of DNA sequence, complementary to each probe, that was present in each of the DNA samples (see **Figure 1-5**).

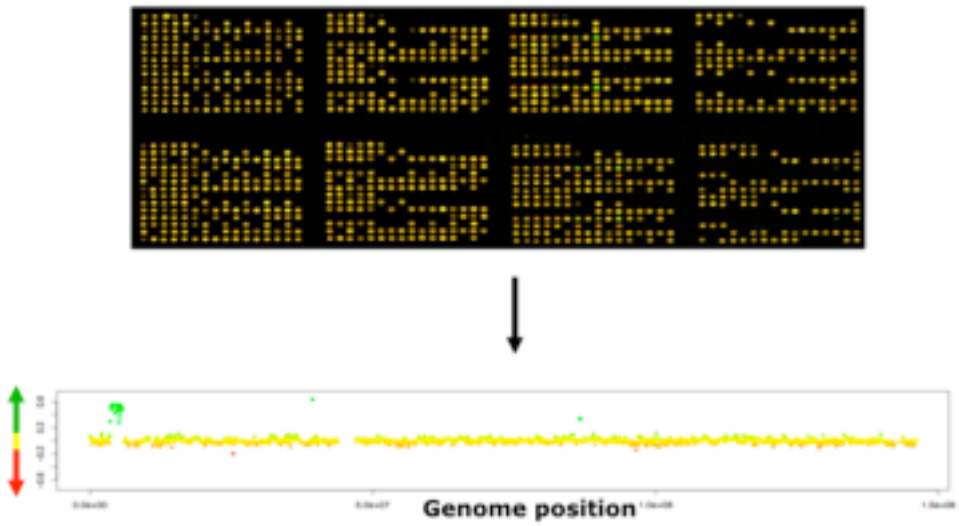


Figure 1-5 Illustration of converting the relative signal intensity at each spot into a log2 ratio.

SNP-Genotyping Arrays

The primary goal of a single nucleotide polymorphisms (SNP) Genotyping array is to detect SNPs within an individual compared to some population [55]. The two main manufactures of SNP chips, Affymetrix and Illumnia, employ slightly difference technologies during array fabrication. Affymetrix uses silicon chips (gene chips) as a solid surface on which they immobilise oligonucleotide probes whereas Illumnia opts for microscopic beads in place of a solid surface. Affymetrix uses a classic photolithographic process, utilising semi-conductor manufacturing techniques, to produce oligo-nucleotide microarrays on silicon (quartz) chips. The most common methods for in situ synthesis of oligo-nucleotide sequences use the process of photolithography (see **Figure 1-6**). A photolithographic process involves the use of light-sensitive masks to block the passage of light to some areas of the solid surface and allowing it to pass through to others.

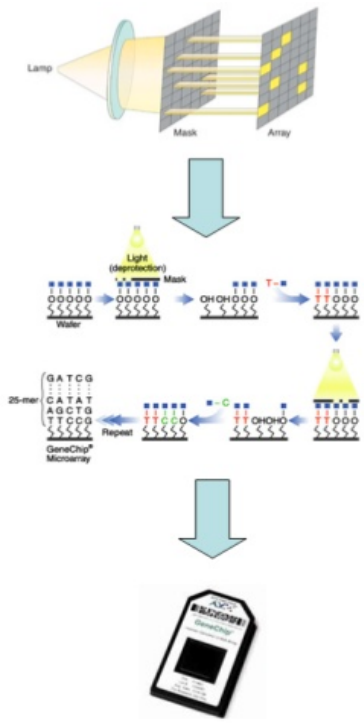


Figure 1-6 Diagram showing the photolithographic process involved in producing an Affymetrix gene chip [affymetrix.com].

The Illumina BeadArray technology uses self-assembling silica beads, 3 microns in size and each of which is covered by thousands of copies of a specific oligonucleotide. These beads are contained in microwells on the surface of either fiber optic bundles or silica slides (see **Figure 1-7**). These microwells then act as the probes to which fluorescently labelled target DNA fragments can hybridise.

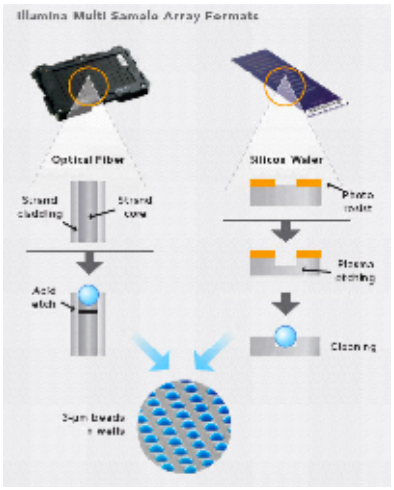


Figure 1-7 Diagram showing BeadArrays using slica slides (left) and fiber optic bundles (right) [Illumina atlas-biolabs.de].

Both Affymetrix and Illumina Genotyping arrays come in a variety of different resolutions and can contain as many as 2 million oligonucleotide probes, they

also come in a variety of formats to allow the processing of several samples on a single slide. They both can contain a combination of SNP ("rs") and CNV ("cnv") probes targeted towards specific locations within the genome. Whether a probe is targeted towards a SNP or a CNV location, all the probes on a Genotyping array can be used to infer the copy number of its genomic location. The Log R Ratio (LRR) and B-allele frequency (BAF) values can be used individually or in combination to detect change point intervals across the genome [56].

Although it may seem attractive to be able to perform both genome-wide genotyping and copy number analysis on a single platform, in reality the performance of genotyping arrays for copy number discovery is severely limited compared to their array-CGH counter parts [57]. Genotyping arrays show limited performance partly due to a general poor response displayed by their probes. This may be due to the fact that the probes are shorter (approx. 40bp) compared to array-CGH probes (approx. 60bp), a weaker intensity strength displayed by the fluorophores used during the labelling reactions or differences in experimental processing. It is also generally observed that the variance of the Log R ratio from a genotyping array is considerably greater than its equivalent (the Log2 Ratio) from array-CGH arrays. As a consequence of both of these factors the signal to noise ratio of genotyping arrays are on average far poorer than that of array-CGH arrays and this has a large impact on the ability to confidently detect change point intervals [58].

1.2.3 Sequence Based Technologies

Capillary Sequencing

One of the first automated sequencing technologies developed was the Applied Biosciences 377 gel electrophoresis DNA sequencing machine. This machine could separate labelled DNA products on a polyacrylamide gel poured between two glass plates. The technology was improved with the introduction of the capillary electrophoresis sequencing machines developed by Applied Biosciences in later years, which used a denaturing flowable polymer to separate labelled DNA products. With the introduction of the ABI 3700, a machine capable of processing 96 samples in one run, DNA sequencing became much more automated [59]. The ease of use, throughput and consistency of results were much improved by this new technology [60]. The Wellcome Trust Sanger Institute extensively used capillary sequencing for the completion of the human genome project [61].

DNA sequencing could be performed by attaching four different fluorescent dyes to ddNTPs, allowing one reaction tube per sample. The reaction is achieved by adding unlabelled primer, buffer, the four dNTPs, the four fluorescently labelled ddNTPs and Taq polymerase (a thermostable polymerase enzyme) into a reaction tube. Cycles of denaturation and amplification are then performed and result in DNA fragments of different sizes having a fluorescently labelled terminator at their 3' end (see **Figure 1-8**). These different sized fragments can then be separated by size and the sequence determined by capturing the emission spectra of the terminator molecule, this approach to sequencing is known as Sanger sequencing [62].

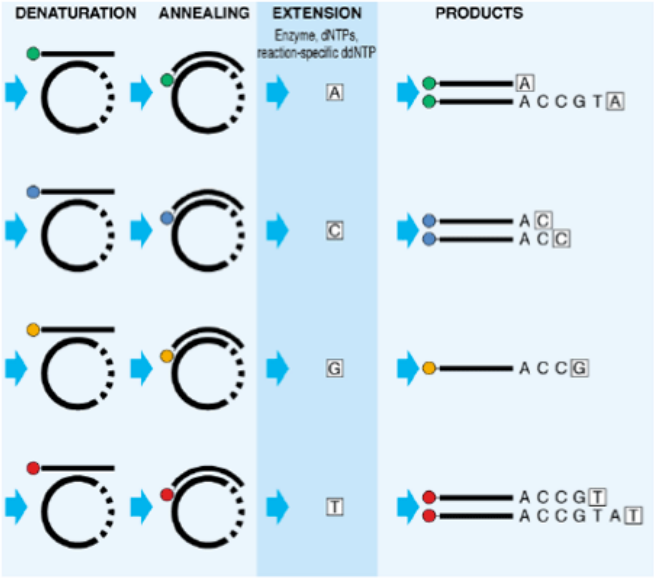


Figure 1-8 Denaturation, annealing and extension processes of Sanger sequencing [appliedbiosystems.com].

Products of the cycles of denaturation and amplification steps are delivered to the capillaries via an electrokinetic injection, where a high voltage charge is applied to the buffer and acts to force the negatively charged DNA fragments into the capillaries.

The DNA fragments are pulled though the capillary by a positive electrode at the end, the polymer inside the capillaries acts to separate the DNA fragments by size based on their total charge. The electrophoretic mobility of the fragments can vary due to various conditions. For example the buffer type, concentration, pH, the run temperature and the amount of voltage can all affect the speed at which the fragments migrate.

As the sample reaches the end of the capillary, the fluorescently labelled DNA fragments move across the path of a laser beam. The laser beam causes the dyes to fluoresce and an optical detection device detects the emission spectra. Since each fluorescent dye emits light at a different wavelength all four colours, and consequently all four bases, can be detected and read from a single capillary injection (see **Figure 1-9**).

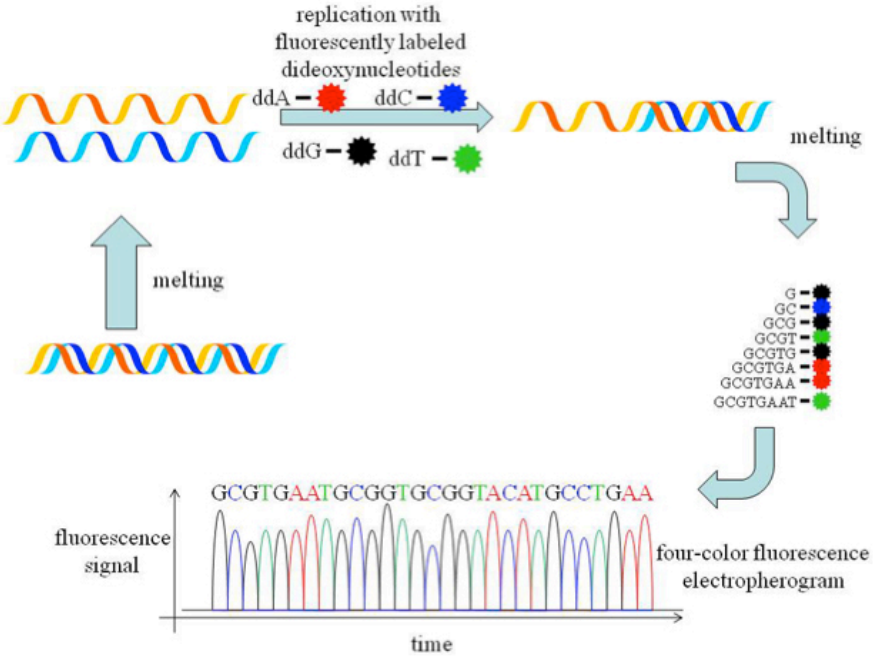


Figure 1-9 Diagram showing the Sanger sequencing process, resulting in a readable sequence of emission spectra [appliedbiosystems.com].

Next Generation Sequencing

There is nowadays a number of different next generation sequencing (NGS) technologies available [63-66]. These include the Illumina Solexa, Roche 454, ABI SOLiD and Pacific Biosciences PACBIO to name just a few. The sequencing technologies and protocols are under constant development and are leading to improvements in the quality and volume of the data that can be produced and to the cost of running an experiment.

The original NGS technology was produced by Solexa and was acquired by Illumina in November 2006 for the price of 600 million dollars. The basic protocol used in a Solexa experiment generates large numbers of unique polymerase generated colonies that can be simultaneously sequenced. These reactions take place in parallel on the surface of a ”flow cell” which is then used as a surface for many thousands of parallel sequencing chemical reactions to occur.

The DNA sample needs to be sheared to the correct size (approx. 800bp). This is achieved using a compressed air device known as a nebulizer. Two unique adapter modules are then ligated to both ends of the fragments. Ligated fragments of the target library size (e.g. 200bp) are isolated and amplified using a number of PCR cycles.

Single stranded oligonucleotides complimentary to the sequences of the adapters molecules coat the surface of the flow cell. The single-stranded ligated fragments from the sample preparation stage are bound to the surface of the flow cell. The reagents necessary for a polymerase extension reaction are applied to the surface of the flow cell. Priming can then occur when the distal end of a ligated fragment ”bridges” to a complementary oligo on the flow cell surface. By

repeating a denaturation and extension cycle amplification of single molecules in millions of unique locations across the flow cell surface can occur (see **Figure 1-10**).

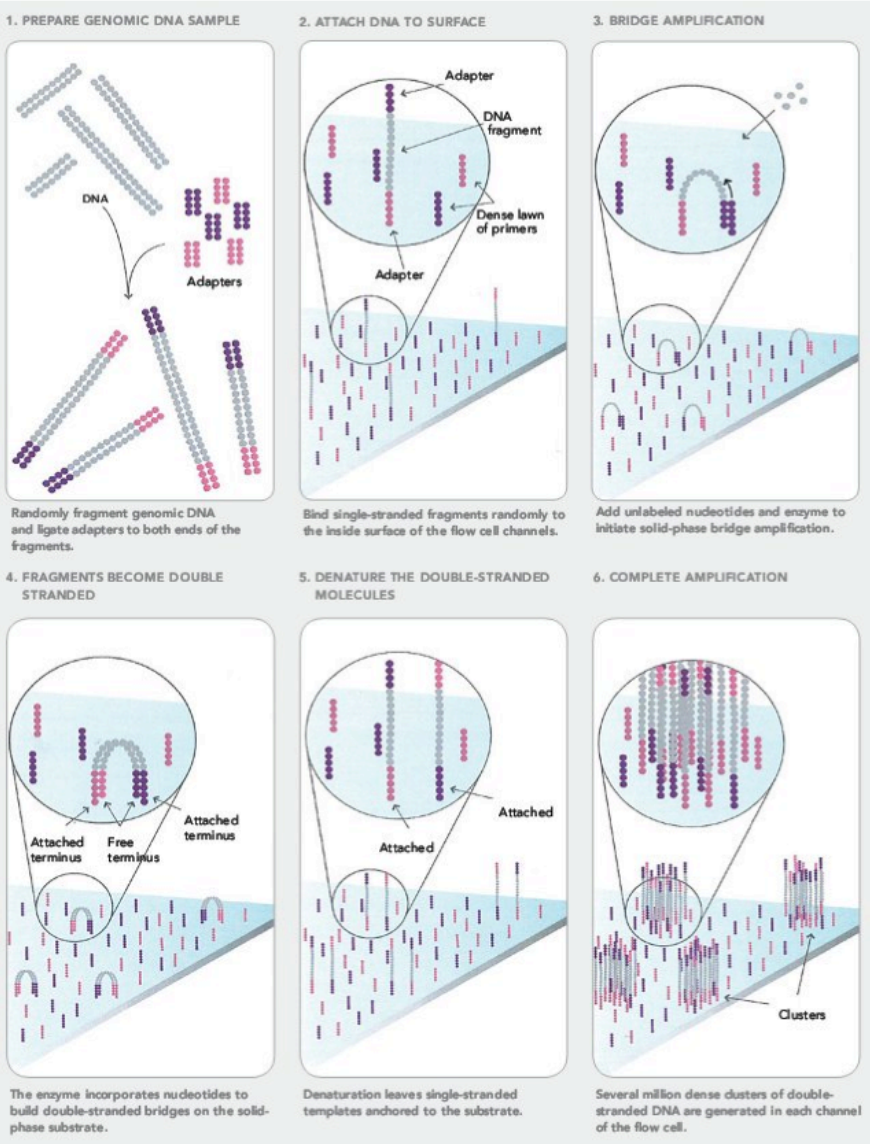


Figure 1-10 Diagram sample preparation, bridge amplification and denature and extension cycles [seqanswers.com].

Automated cycles of extension and imaging can occur by loading a flow cell containing millions of unique clusters into the sequencer. The incorporation of a fluorescently labelled nucleotide, followed by high resolution imaging of the flow cell can now occur. The first image captured represents the data collected for the first base of the sequence. Signals above the background identify the physical location of a cluster and the precise fluorescent emission identifies which of the four bases was incorporated at that position. Each nucleotide base is labelled with a different coloured fluorescent dye that allows the sequence of each cluster to be read. A cycle is then repeated to generate a series of images representing a single base extension at each specific cluster. The exact sequence of each cluster

is derived using an algorithm to detect the emission spectrum across all of the images for each cluster (see **Figure 1-11**).

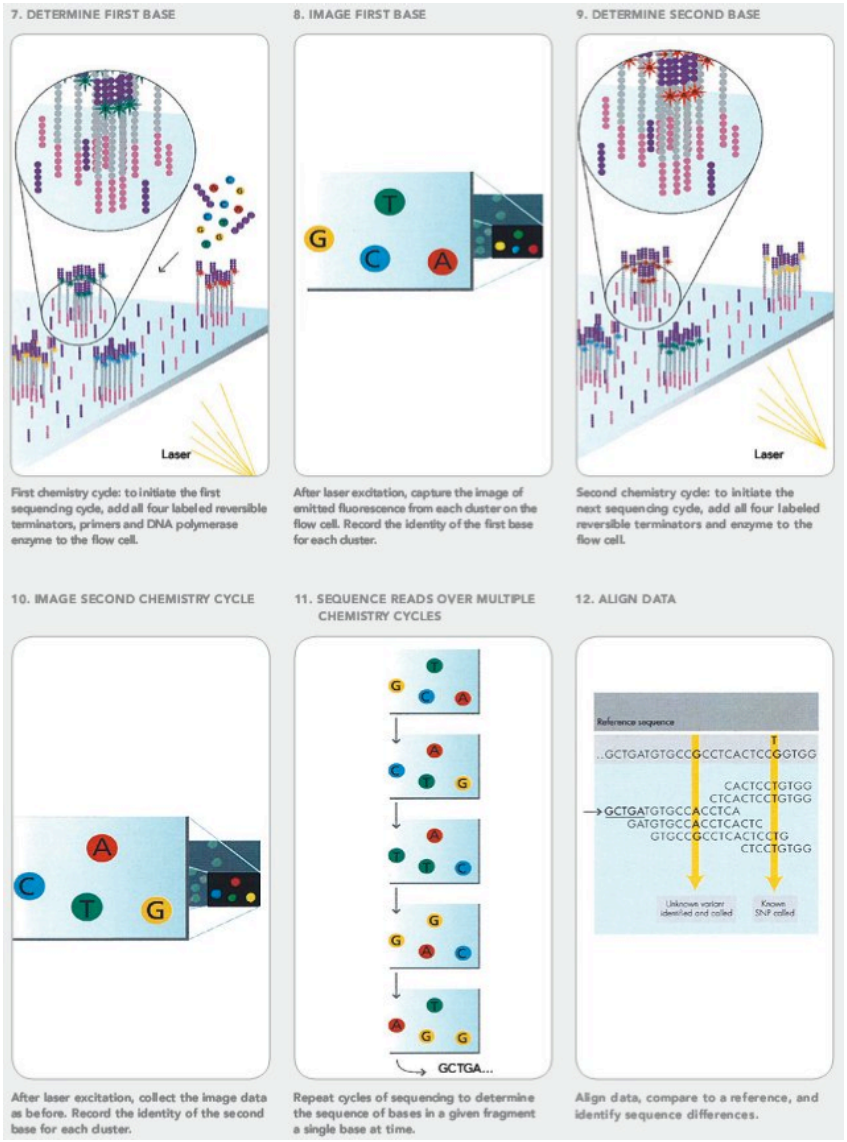


Figure 1-11 A diagram showing data collection, sequence determination and alignment [seqanswers.com].

Each type of sequencing technology has different performances in various areas, for example, the length of reads, the density of reads and the amount of noise reduction they produce can vary across platforms.

The original Solexa machines could only achieve short reads (approx. 30-60bp) but could achieve very high throughput. The 454 from Roche could produce much longer read (approx. 400bp) but at a far lower throughput rate [64]. The newly emerging PACBIO from Pacific Biosciences promises to deliver the ability to tune these types of parameters to suit individual purposes [67-69]. For example, they claim to be able to produce a relatively small number of long reads and a large number of short reads across a genomic location using a method called strobe sequencing [70].

Striking the correct balance between these types of parameters are what the sequencing companies and R&D teams are working hard towards and there seems to be almost monthly releases of new and improved protocols. However, our ability to extract the maximum amount of useful information from these data using analytical methods is lagging behind at an ever-increasing rate [71]. Although a fair amount of these data can be utilised already there are still a lot of reads that are currently not interpretable [72] however, as the sequencing data quality improves so analytical methods also become robust. There is however a time cost associated with data improvement where either existing analytical methods need to be modified or new ones created [73].

For copy number information there are currently two useful properties of the sequence data, the incorrect mapping of read pairs and the read depth at each genomic location. Although copy number from NGS is receiving a lot of attention from the community [27, 74] the ability to accurately assign copy numbers is hampered by a number of factors. Among these, the differences in platform performances, technical noise, normalisation strategies and analytical approaches all conspire to make the interpretation of copy number from NGS data a challenging area of research.

1.3 COPY NUMBER DATA

1.3.1 Single Colour Data

Single-channel (single colour) microarrays such as the Affymetrix gene chip or Illumina BeadArray provide a single distribution of intensity values from each probe on the array. As a result single-channel (genotyping) arrays normally use a large reference set specific to each array format to normalise the intensity values against. This allows the relative intensity levels of each probe to be compared to a population mean and thus provide an estimate of the amount of genetic material present, for each probe, in the sample compared to the mean of a population. High-density SNP genotyping platforms (single colour microarrays) produce an intensity value for each allele of a given SNP marker. The careful analysis of the signal intensities produced by these arrays can be used to identify regions with multiple SNPs that support deletions or duplications at specific locations throughout the genome [75, 76].

Log R Ratio

The Log R Ratio (LRR) is a normalized measure of the total signal intensity for the two alleles of a SNP marker. Specifically, it is the log2 of the total signal from both alleles of a SNP marker divided by the average total signal from both alleles of the same SNP marker from the reference set.

Let sX and sY equal the signal intensity observed in SNP allele (A) and SNP allele (B) from the test sample, respectively and let rX and rY equal the average signal intensity observed in SNP allele (A) and SNP allele (B) from the reference set, respectively. Finally let N equal the total number of SNP markers.

We then calculate the LRR values thus:

$$LRR_n = log_2 \frac{sX_n + sY_n}{rX_n + rY_n} \quad [1\text{-}1]$$

with *n* from 1, 2, ..,*N*.

This transformation is often written in this form:

$$R = X + Y \quad [1\text{-}2]$$

and

$$LRR = log_2 \frac{Robserved}{Rexpected} \quad [1\text{-}3]$$

where *Rexpected* is computed from linear interpolation of canonical genotype clusters obtained from a set of reference samples [76].

B-Allele Frequency

The B Allele Frequency (BAF) is a normalized measure of the allelic intensity ratio of two alleles.

The BAF is a measure of the allelic intensity ratio: When a deletion CNV is present, BAF values cluster around 0 or 1 but are absent around 0.5; when duplication is present, BAF values cluster around 0, 0.33, 0.67, and 1, reflecting the AAA, AAB, ABB, and BBB genotypes, respectively.

The θ value measures the relative allelic intensity ratio of two alleles and is calculated as $\theta = \arctan(Y/X)/(\pi/2)$ which ranges from 0 to 1. The BAF refers to a normalized measure of relative signal intensity ratio of the B and A alleles:

$$BAF = \begin{cases} 0, if\ \theta < \theta_{AA} \\ 0.5(\theta - \theta_{AA})/(\theta_{AB} - \theta_{AA}), if\ \theta_{AA} \leq \theta < \theta_{AB} \\ 0.5 + 0.5(\theta - \theta_{AB})/(\theta_{BB} - \theta_{AB}), if\ \theta_{AB} \leq \theta < \theta_{BB} \\ 1, if\ \theta \geq \theta_{BB} \end{cases} \quad [1\text{-}4]$$

To search for regions of LOH and UPD in SNP genotyping data the BAF is often used. BAF is a value between 0 and 1 and represents the proportion contributed by one SNP allele (B) to the total copy number (see **Figure 1-12**).

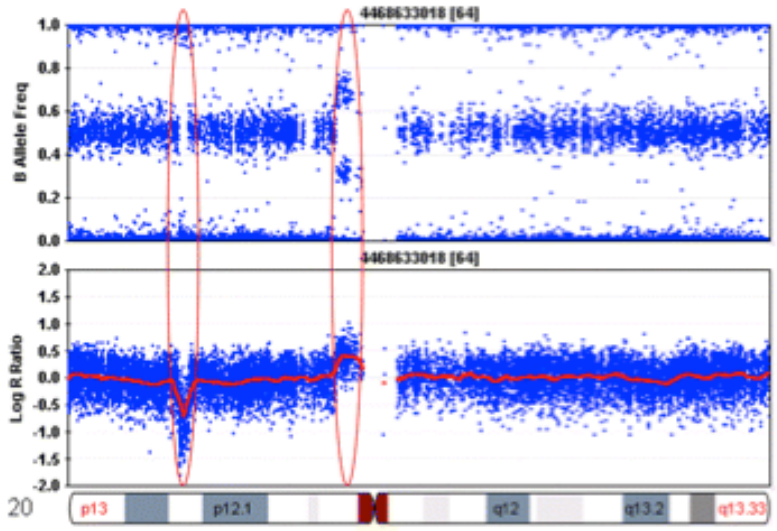


Figure 1-12 Illustration of the BAF (upper panel) and LRR (lower panel) values in chromosome 20 of a person with large (1 Mb) deletion and duplication [scientificprotocols.org].

1.3.2 Dual Colour Data

Two-channel (dual colour) microarrays are normally hybridized using cDNA prepared from two different samples labelled with two different fluorophores, usually Cy5 and Cy3 [77]. The cytosine3 dye has a fuorescence emission wavelength of 570nm (green) and the cytosine5 an emission wavelength of 670nm (red).

The two differentially labelled samples are combined in the presence of buffer and applied to a single microarray such that competitive hybridisation can occur. After an incubation period the microarray can be scanned using a laser beam to promote excitation of the fluorophores at the defined wavelength and the relative intensities measured.

Intensity Distributions

The extracted intensities of each fluorophore, for each probe on the array can then be normalised in a number of ways to achieve similar ranges and scales and remove any bias between the two distributions.

Log2 Ratio

These intensity distributions can then be transformed into a Log2 Ratio (L2R), which reflects the relative amount of genetic material present in each of the two samples for each of the probes on the array.

Let *c3* and *c5*, equal the signal intensity extracted at the 570nm and 670nm wavelengths, respectively.

Finally let *N* equal the total number of probes on the array.

We then calculate the Log2 Ratio values thus:

$$L2R_n = log_2 \frac{c3_n}{c5_n} \quad [1\text{-}5]$$

with *n* from 1, 2, ..,*N*.

The L2R values can then be normalised in a number of different ways to remove some further technical biases. It is then possible to use the L2R values as a direct estimation of the relative difference in abundance of genetic material, between the two samples, at all of the genomic locations covered by the array.

It should be noted that by using this two-sample approach it is not possible to get an absolute copy number estimation for either of the samples, rather only the relative copy number between the two samples is available. One way to overcome this is to use a ”pooled” reference DNA to label and apply to the array in place of one of the samples [78]. Using this approach it is possible to obtained the true copy number, of a given sample, at a given genomic location, compared to the population mean. It is generally believed that as few as 100 ”normal” DNA samples will provide a relatively accurate estimation of the population mean.

1.3.3 Sequence Data

The information that can currently be utilised from sequence data to estimate copy number (structural variants) comes in the form of either read-pairs or read-depth.

Read Pairs

Paired-end sequencing technology can be used to detect structural variants [79]. Suppose a clone library for an individual with a clone size of 150kb has been prepared. When these clones are sequenced and mapped back to the reference genome, the ends of the clones would be expected to map at an approximate distance of 150kb away from each other in a forward reverse direction. Therefore, if any read pair that does not map in this manner it could be an indication of a structural rearrangement (see **Figure 1-13**).

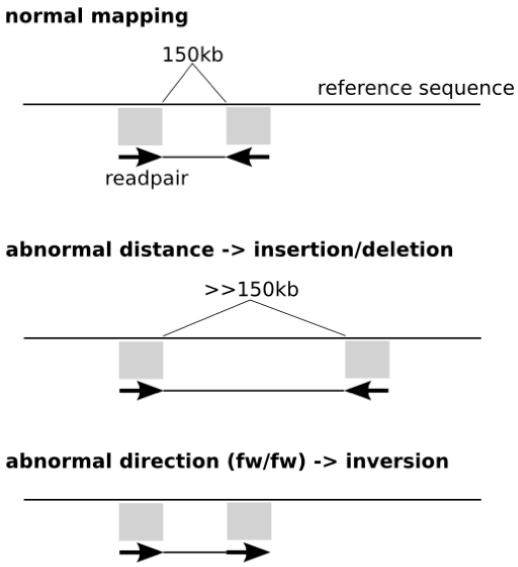


Figure 1-13 An illustration of incorrectly mapping read pairs – [saaientist.blogspot.com].

As illustrated above (see **Figure 1.13**) there are several possible outcomes when mapping read pairs back to the reference genome. Firstly the pairs could map back at the correct distance from one another and this indicates that there are no structural variants in this region for the sample compared to the reference genome.

Secondly there is the situation where the read pairs map at an abnormal distance from each other. This indicates that the variant is either a deletion or an insertion at this position for the sample compared to the reference. If the pairs map a shorter distance away from each other it is evidence for a deletion whereas if they map too far apart it is evidence for an insertion. The final possibility is given when the read pairs map to the reference genome in the wrong orientation relative to each other. This type of read pair mapping discrepancy can indicate the presence of an inversion.

Read Depth

In read depth data from NGS technology there is seemingly the perfect data set for copy number estimation [80]. The amount of sequenced product produced is, in theory, directly related to the amount of starting genetic material in the sample. So if an individual contains more or less of a particular genetic sequence the proportion of this sequence should be reflected in the amount of sequenced product produced from the sequencing reaction. Thereby, it would be possible to simply count the number of reads for each of the sequences produced to get an estimation of the number of copies of that genetic sequence the sample contained. Indeed, this is possible and methods have been developed to utilise read depth in much the same way as the log2 ratio from an array-CGH experiment [81].

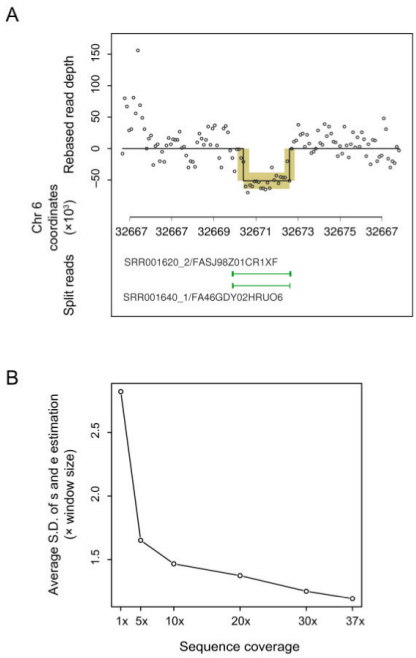


Figure 1-14 An illustration of using read depth for copy number discovery [81].

Short reads from an individual were mapped to the human reference genome and counted in a 100bp non-overlapping sliding window to produce the read depth data (see **Figure 1-14**). The counts were then centred on their mean and [81] applied their MCMC sampler to detect change point intervals from the data. One of the detected segments is shown above, it is a 2653bp deletion locus. This demonstrates the potential read depth has for detecting copy number variation from NGS data.

However, read depth data still has various associated problems; there are a number of biases (for example a GC bias), which need normalisation strategies to avoid. Additionally variable read qualities, achieved sample depths and constantly changing experimental protocol have a large impact on the utility of the data and on the associated analytical methodologies.

1.4 COPY NUMBER ANALYSIS METHODS

1.4.1 Array-CGH Data Normalisation

Dye Bias

It is well established that a bias exists between the two fluorophores normally used in array-CGH (Cy5 and Cy3) [82] due to differential efficiency in enzymatic labelling. This bias manifests as a difference between the two resulting intensity distributions and if left uncorrected is highly likely to skew the ratio calculated during downstream analysis. Over the years there have been a number of different approaches taken for dye-bias removal including both linear and non-linear methods [83]. These methods were primarily developed for the normalisation of expression microarrays and have more recently been applied to array-CGH microarrays [84].

The most commonly used and widely trusted dye bias normalisation methods tend to be based on lowess (locally weighted regression) techniques. However, standard regression techniques can be sensitive to outliers, which are common features of array-CGH data. As a result, lowess-based dye bias normalisation approaches normally use robust regression techniques, becoming computationally intensive and severely limiting if applied to microarrays comprising more than three hundred thousand probes [85].

Normalisation methods based on fitting smoothing spline functions have most often been seen in relation to between-chip quantile normalisation of microarrays [86] and, although promising work has been carried out previously [87], are still seldom applied to two-colour array-CGH microarray data. One possible explanation for this is that when fitting a spline curve only a subset of data is used and without appropriate action can lead to inaccurate data adjustment when applied to complex datasets [88].

More recently we published a method (aCGH.Spline) using outlier exclusion and posterior interpolation allowing robust fitting of the two intensity distributions, independently towards their geometric mean using cubic spline interpolation [89].

The cubic splines used in this study were of the form:

$$S_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + (x - x_j)^3 \qquad [1\text{-}6]$$

and if a given set of coordinates (knot points) are:

$$\{(x_0, y_0), (x_1, y_1), \dots (x_n, y_n)\} \qquad [1\text{-}7]$$

the required set of *n* splines are:

$$S(x_i) = y_i, i = 0, \dots, n - 1 \qquad [1\text{-}8]$$

These were then computed, using Gaussian elimination, subject to the usual derivative constraints for the splines. **Figure 1-15** shows microarray normalisation plots (MAplots) for aCGH.Spline compared to a number of other dye bias normalisation approaches.

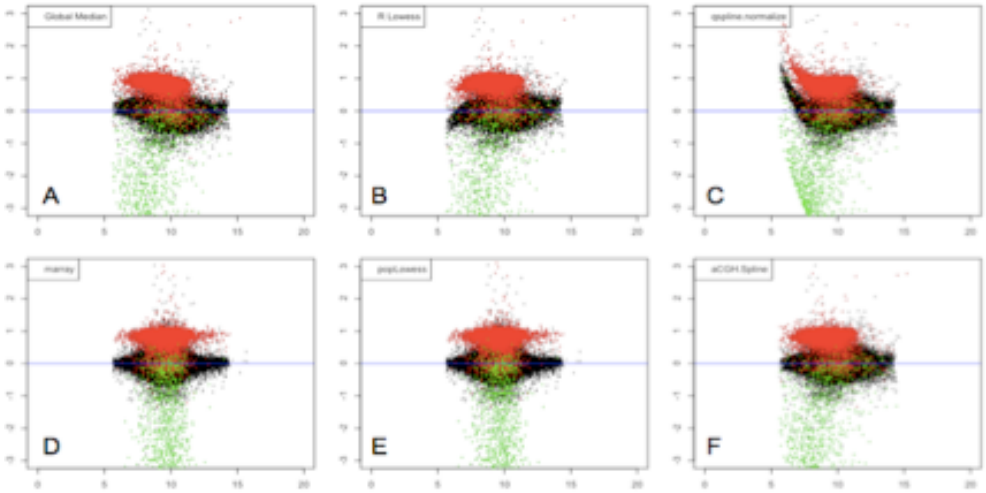


Figure 1-15 MA-plots showing normalisation quality of sample 7 when using 6 different normalisation methods. Panels showing one 244k custom tiling Agilent array under six different normalisation methods, autosomes (black), chromosome X (red), chromosome Y (green). (A) Global median normalisation. (B) Standard lowess function in R. (C) qspline.normalize method from the affy biocondutor package. (D) printTipLoess method from the marray bioconductor package. (E) popLowess method. (F) aCGH.Spline method.

It is clear that, when using only a global median normalization, dye bias removal will be compromised due to not accounting for any non-linear effects (**see Figure 1-15 A).** Moreover, when using either the global spline fitting method 'qspline.normalize' or the locally weighted regression method 'R lowess', where no prior outlier exclusion is applied we can observe an inaccurate adjustment of both outlier and non-outlier data points (**see Figure 1-15 B and C**). When using the robust regression methods, 'marray' and 'poplowess', the adjustment of data points are not adversely affected by outlier values; however, the adjustment itself results in the distribution of data points being forced towards a normal distribution (**see Figure 1-15 D and E**). With aCGH.Spline, by excluding outlier values prior to spline fitting and correcting them by posterior interpolation the distribution of data points closer to the true distribution (**see Figure 1-15 A and F**) but with most of the dye bias removed (**see Figure 1-15 F**).

Probe Bias

One historical approach to removing the dye bias from array-CGH data and improving the data quality was to perform the so-called dye-swap designed experiment [84]. A dye-swap design involves running two separate arrays for each hybridisation where the dye used to label the two DNAs being compared is switched between the first and second hybridisations. Although it can be effective, this is not a satisfactory solution as it involves combining two independent array profiles and doubles the cost of the experiment. On top of that the data quality of each experiment is likely to differ and needs to be combined to remove the effect of the dye bias.

One alternative that has been considered previously [90] is the self-self normalisation procedure. A self-self array-CGH experiment is defined as running the same sample, differentially labelled, against itself on an array. The theory was that by running the same sample against itself on the array, the differences that were seen between the two resulting intensity distributions would be due to the dye bias. Interestingly the differences seen when running a self-self experiment cannot be fully explained by the dye bias and is not an effective method for removing it. Of course, the degree of dye bias present is likely to differ between replicate experiments and does not show a linear relationship [89]. However, the type of noise detected by a self-self experiment can be explained and utilised to remove a different type of technical bias that exists within microarray-based data. Theory dictates that in a self-self experiment every probe should report with a log2 ratio of zero, for no copy number change, however in practice this is not the case even when correcting for any dye bias and each probe will report with a log2 ratio either above or below zero (see **Figure 1-16**).

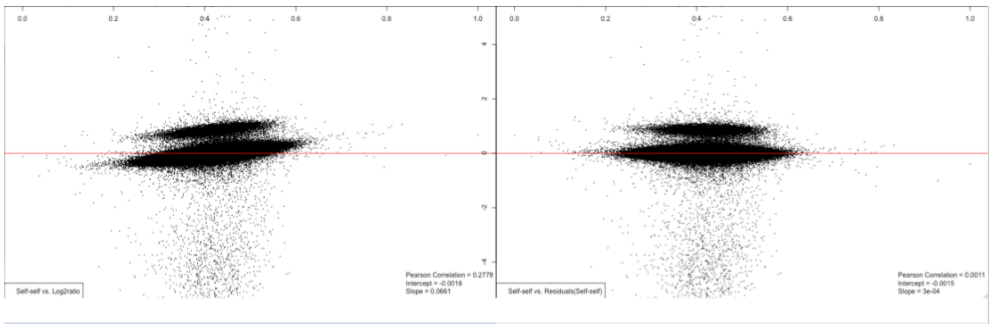


Figure 1-16 Left - The correlation between the derived self-self correction values and an array-CGH array. Right - The correlation between the derived self-self correction values and the residuals of a linear regression against an array-CGH array.

This type of noise can be described as array specific and is an indication of systematic differences in probe efficiencies across the entire array. By running a number of self-self calibration data across each array platform being used it is possible to start to estimate the average behaviour of each and every probe on

the array. By using these values during normalisation it is possible to remove this systematic noise and greatly reduce the noise levels across an array.

Wave Bias

Auto-correlation was first linked with array data to describe the type of spurious false results occurring in association with non-random probe placement during array manufacture. Computer simulations were used to show that large spatial biases across an array, those that could result from microarray hybridisation, could exclusively account for spurious correlations in the data [91]. Moreover the presence of an auto-correlation or genomic wave effect has been described as a specific type of systematic noise within the log2 ratio values of array-CGH data [92]. This type of systematic noise appears to be present on most microarray platforms and without proper adjustment has the potential to cause large numbers of false discoveries.

When using BAC large insert clone array a high correlation between the GC content of each BAC clone and the log2 ratio it reported was described previously [92]. This observation suggested that there could be a GC dependent bias during either DNA labelling or array hybridisation. Nevertheless, we were able to apply a GC based correction, using linear regressions, to BAC based array-CGH data and remove most of the genomic wave effect [92]. Here this approach was extended to operate on oligo-nucleotide arrays (see **Figure 1-17**). Following, others have developed similar methods based on the GC content of probes to correct for the genomic wave effect seen on Affymetrix SNP arrays [93]. They also observed that the magnitude of the wave effect could be linked to both the concentration and quality of DNA being applied to the array. The most recent method to be developed using a GC correction is the array-CGH waves correction algorithm (WACA) [94].

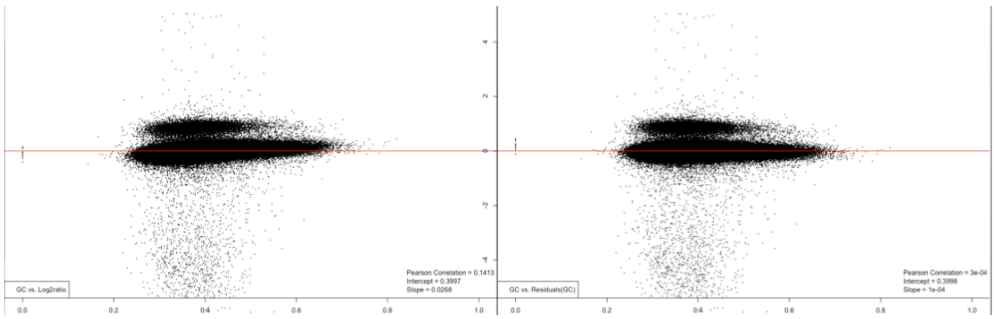


Figure 1-17 Left - The correlation between GC correction values and an array-CGH array. Right - The correlation between GC correction values and the residuals of a linear regression against an array-CGH array.

The combination of these three normalisation strategies can be highly effective for the de-noising of array-CGH log2 ratio data. **Figure 1-18** shows the effect of applying both a probe bias and a wave bias normalisation on the dye bias normalised log2 ratio values from a single array-CGH array.

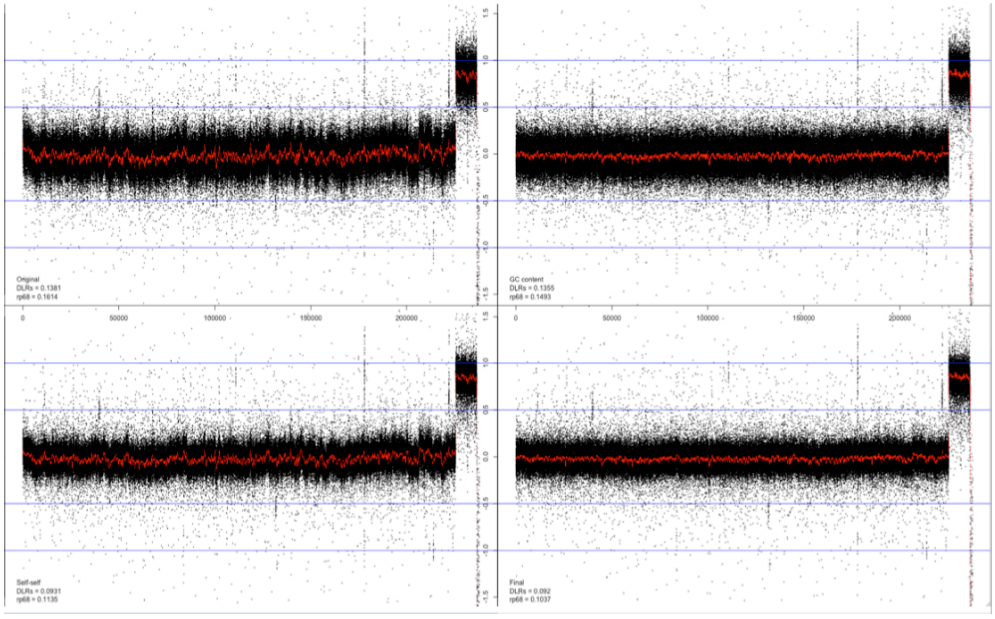


Figure 1-18 The effect of removing both the probe bias and the wave bias from an array-CGH array. Top left-original data values, Top right-data values after removing the probe bias, Bottom left-data values after removing the wave bias, Bottom right-data values after removing both the probe and wave bias. Red line is a running median spanning 301 probes.

Due to the special properties of wavelets they are most often used to extract information from data such as audio signals or images. Wavelets have a rich history in image analysis and signal processing and have been successfully used in the de-noising of frequency-based data for some time [95]. The use of wavelet transform methods has previously been applied to the normalisation of the dye bias in array-CGH data but, for the most part has not been seen often in relation to microarray data normalisation [85].

More recently a dual-tree-complex-wavelet approach has been described to remove scanner noise from microarray images derived from Affymetrix GeneChip arrays [96]. Wavelets, having been highly useful in signal processing and image analysis, are often used to extract frequency-based information from unknown signals. No model exists for the genomic wave and it shows non-uniform (non-stationary frequency) characteristics, giving different patterns of oscillation between replicate experiments. Surprisingly the use of wavelet transforms to correct for the genomic wave effect is yet to be fully explored. There is one relatively recent method described for the de-noising of BAC large insert clone arrays showing small-scale genomic rearrangements [97]. However, the use of such methods is not commonplace and has yet to show its true potential to the microarray community. Due to the special properties of wavelets, we were able to use them as functions to operate on the log2 ratio distribution of array-CGH data and extract different scale components (octaves) relating to the genomic wave (see **Figure 1-19**).

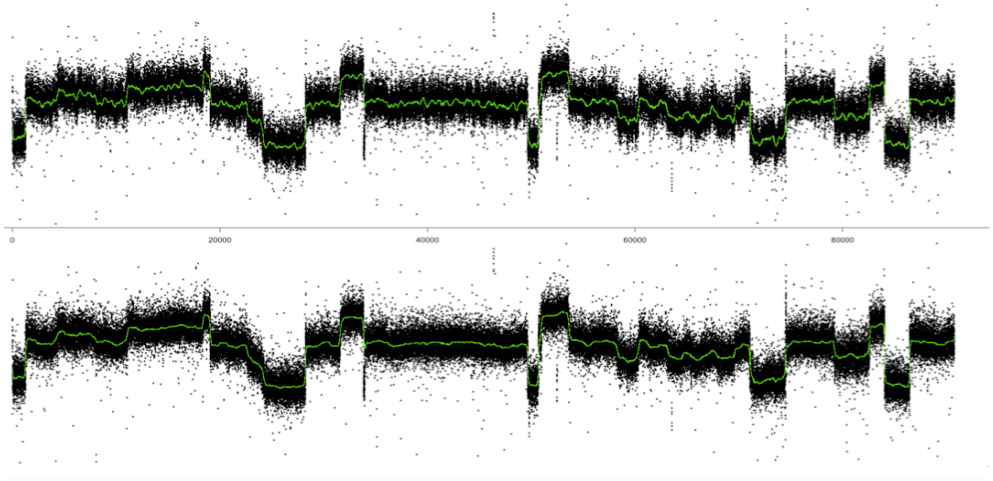


Figure 1-19 An example of using the wavelet based method to remove the wave bias from an aneuploidy sample run on an array-CGH array. The green line is a running median spanning 301 probes.

By developing the method it was possible to both increase our understanding of the genomic wave and to remove its presence from microarray based data without the need for GC content measures (method not published). There were however a few complications surrounding the application of wavelets and it was an absolute priority to ensure that the data were not corrupted in any way. As array-CGH data are complex by nature and can contain large numbers of outliers with several distinct distributions of log2 ratio values within one experiment, the correct use of appropriate filters and the development of methodologies to account for these features were essential.

1.4.2 Change-Point Detection Theory

Whenever we talk about copy number variation within a genome, in reality what we are really talking about is change-point detection. In other words, the point at which data features change for one state to another (in this case different copy number states). Formally change point detection is a statistical analysis to search for changes in the probability distribution of a stochastic process or time-series.

Most of the currently available methods for copy number discovery, with the exception of read-pair data, result in the generation of datasets that can be thought of as time-series. These datasets provide discrete measurements indexed by a time like parameter (genomic position in this case). Although there are many different approaches concerned with change point detection, at the root of all these methods one can find a hypothesis test; given some characteristics of the dataset, is the measurement y taken at time x significant.

Threshold Based Methods

A simple yet often used, frequentist approach to change point detection is to use some threshold above which measurements within a dataset are considered to be significant. This threshold could be a static predefined value or better it could be dynamic and dependent on the dataset itself.

A simple way to illustrate the use of dynamic thresholds for change point detection is to consider that our dataset shows normal, Gaussian "white" noise, or that some measure of the normally distributed noise within the dataset can be obtained. In practice this is normally more difficult since real data are complex in nature and often not normally distributed however, for the purpose of this illustration let's assume that it is.

Since we are now working with normally distributed data we can start talking about a bell shaped frequency curve which is centred around the mean (see **Figure 1-20**). As such we can now say with confidence that the probability that a measurement is above the mean is 50% and the probability that a measurement is below the mean is also 50%.

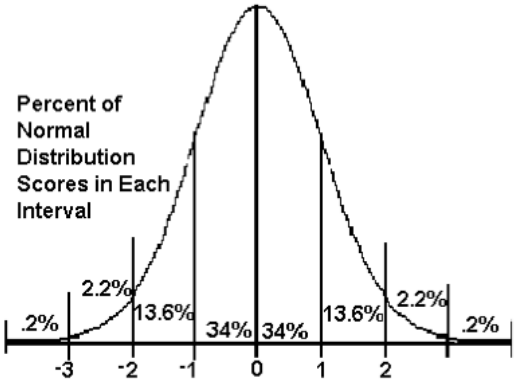


Figure 1-20 The Frequency Distribution of Standard Normal Data - The numbered line is marked in terms of the mean and the standard deviation [mathnstuff.com].

Furthermore, we can define a function that perfectly describes the frequency of a set of measurements using only the mean and standard deviation.

The function is defined as:

$$f(x,\mu,\sigma) = \frac{1}{(\sigma\sqrt{2\pi})} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \qquad [1\text{-}9]$$

where μ is the distribution mean and σ is its standard deviation. The distribution is said to be standard normal if $\mu = 0$ and $\sigma = 1$.

Hypothesis Testing

One of the more important statistical tools for making decisions is that of Hypothesis Testing. The role of hypothesis testing is to decide of one from a set of possible outcomes (hypotheses) as a result of data measurements. A test result is considered as statistically significant if it has been predicted according to some significance level as being unlikely to have occurred due to sampling error (data noise) alone.

For detecting copy number variation using time series (array based) data these decisions are normally based on data measurements displaying a continuous

distribution (log2 ratio values). The goal is to detect points at which these data measurements are likely to have changed from one copy number state to another (change points locations). It is possible to consider that the log2 ratio distribution from any array-CGH experiment is a mixture distribution with a finite set of mixture components (relating to different copy number states). These distributions can sometimes contain relatively large numbers of differences in copy number state (see **Figure 1-19**) and it can be difficult to assign the 'normal copy number state' (copy number 2). One approach that can be used for assigning the baseline (null) distribution is to use the mixture component that forms the greatest proportion of the population as a whole.

Once the baseline (null distribution) has been defined it then becomes necessary to define a set of criteria by which one can test the probability of individual data points belonging to that null distribution. For illustration purposes let us set up a simple threshold test where we assume normal Gaussian (white) noise and define an outlier detection threshold as the mean ± 3 times the standard deviation of the log2 ratio values from the null distribution. Additionally, to aid segmentation, let us add a time dependent parameter where we require at least three consecutive data measurements to exceed or remain within the threshold for defining the starting and ending points for an outlier segment (change point interval) respectively.

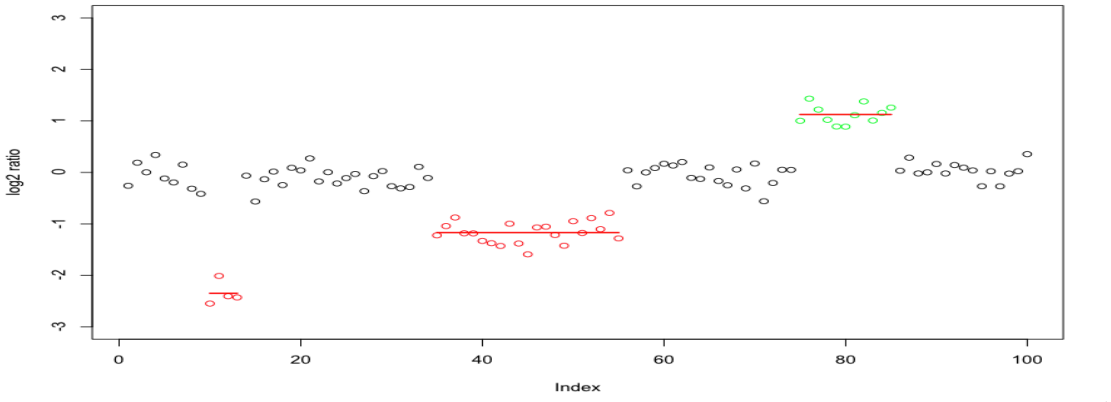


Figure 1-21 Synthetically generated data showing three change point intervals detected using a simple threshold based approach.

Above (see **Figure 1-21**) is the result of applying these threshold based rules in a forward direction to some synthetically generated data. The data contains one hundred normally distributed data points with a mean of zero and standard deviation of 0.2. Three synthetic change point intervals were added with lengths of 4, 21 and 11; and means of -2.3, -1.2 and 1.1 respectively. The detection threshold of 0.6 (see above) was used for defining data points outside of the null distribution and the three consecutive data point rule was used to define the starting and ending position for each change point interval. Using this approach all three known (synthetic) change point intervals were correctly detected (see **Figure 1-21**). There were no false detections made and there was no one single data point predicted to be outside of the null distribution other than those inside the synthetic change point intervals.

Segmentation Based Methods

Segmentation methods are a particular type of change point detection that is not only concerned with finding discrete change point locations (or breakpoints) within data but in defining change point intervals (or segments). In reality, these methods are rather similar to standard change point detection methods, however they also include the definition of for how long a change was observed (i.e. its stating and ending point across the data index measure).

There are a large number of different segmentation based change point detection methods but a well known, simple and good example comes in the form of CUSUM (or cumulative sum control chart), which was originally proposed by E. S. Page [98].

As its name implies CUSUM charts are constructed by calculating a cumulative sum across the data. These cumulative sums can extend across all of the data measurements, be limited to particular sized segments or be started and ended defined by a certain set of criteria. For the case where we extend across the entire data:

Let X equal the data measurements, let N equal the number of data measurements, and let S equal the cumulative sum.

First we calculate the mean:

$$\bar{X} = \frac{(X_1 + X_2 +, \dots, X_N)}{N} \qquad [1\text{-}10]$$

Then start by setting $S_0 = 0$, and follow by:

$$S_i = S_{i-1} + (X_i - \bar{X}) \qquad [1\text{-}11]$$

with i from 1, 2, ..,N.

The results of this method can be interpreted in a very intuitive manner.

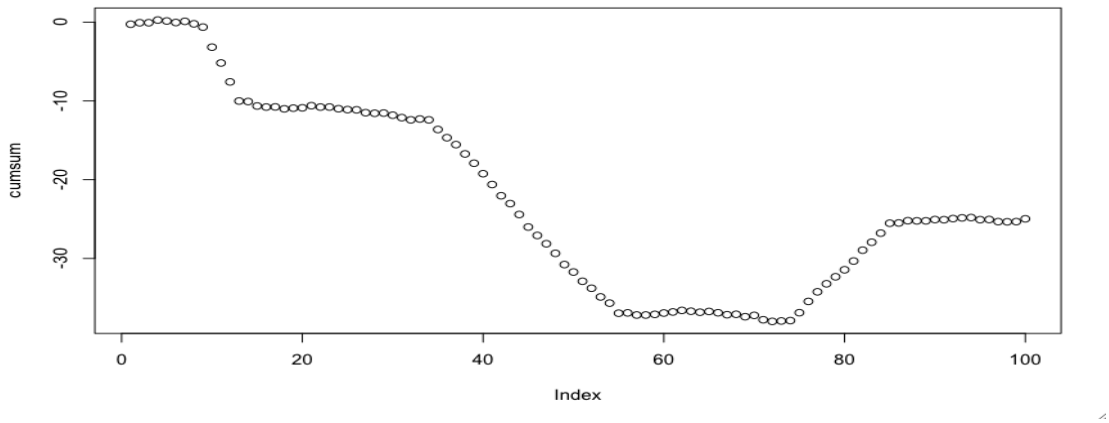


Figure 1-22 Cumulative sum of synthetically generated data containing three known change point intervals

Above (see **Figure 1-22**) we show the cumulative sum across the data measurements from Figure 1-21 assuming a mean of zero. As more values are added to the cumulative sum, if the majority of measurements were positive with respect to the mean the sum will steadily increase. As a result, a segment of the CUSUM chart that displays an upward slope indicates a period of time where the measurements tended to be above the mean. The inverse can also be stated for measurements that tended to be below the mean.

The locations of the known (synthetic) change point intervals are clearly observed in the above plot (see **Figure 1-22**). The two synthetic deletions (starting at indexes 10 and 35 respectively) show the expected downward slope whereas the synthetic duplication (starting at index 75) showing the expected upward slope across the CUMSUM chart.

State-space Models

State-space models define a probability density over real-valued observation vectors $\{Y_t\}$ that are assumed to have been generated from a sequence of hidden state vectors $\{X_t\}$. The hidden state vectors obey the Markov independence property and the model specifies that at any time step the observation vector is statistically independent from all other observation vectors (see **Figure 1-23**).

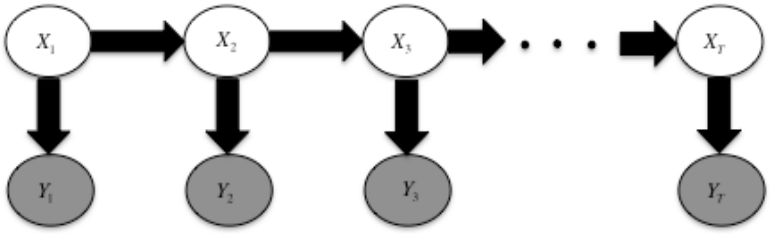


Figure 1-23 Directed acyclic graph (DAG) showing the hidden state vector X_t in white and the real-valued observation vector Y_t in grey.

In the above graph (see **Figure 1-23**) each node is conditionally independent from its non-decendants give its parents and the output vector Y_t is conditionally independent from all others given state X_t. Furthermore, X_t is conditionally independent from $X_1, \dots, X_{t-2}$ given X_{t-1}. The joint probability of the sequences of states X_t and observations Y_t can therefore be formulated thus:

$$P(\{X_t, Y_t\}) = P(X_1)P(Y_1|X_1)\prod_{t=2}^{T} P(X_t|X_{t-1})P(Y_t|X_t) \qquad [1\text{-}12]$$

Assuming the transition and output functions of the model are linear and time-invariant and the distributions of the state and observation variables are multivariate Gaussian, the state transition function is:

$$X_t = AX_{t-1} + w_t \quad [1\text{-}13]$$

where A is the state transition matrix and w_t is zero-mean Gaussian noise. The output function is:

$$Y_t = X = CX_t + v_t \quad [1\text{-}14]$$

where C is the output matrix and v_t is zero-mean Gaussian output noise with covariance matrix R; $P(Y_t|X_t)$ is therefore also Gaussian:

$$P(Y_t|X_t) = (2\pi)^{-D/2}|R|^{-1/2}exp\left\{-\frac{1}{2}(Y_t - CX_t)'R^{-1}(Y_t - CX_t)\right\} \quad [1\text{-}15]$$

where D is the dimensionality of the Y vectors. To model situations where the observation vector can be separated into input (predictor) and output (response) variables we can include a state-transition function:

$$X_t = AX_{t-1} + BU_t + w_t \quad [1\text{-}16]$$

where U_t is the input observation vector and B is the input matrix.

The problem of estimating the posterior probabilities of the hidden variables given the sequence of observed variables in known as *inference* and can be separated into three main methods, filtering, smoothing and prediction. During filtering a recursive algorithm known as the 'Kalman filter' is used to compute the probability of the current hidden state X_t given the sequence of inputs and outputs up to time $t - P(X_t|\{Y_1, \dots, Y_t\}, \{U_1, \dots, U_t\})$. For smoothing, the probability of X_t given the sequence of inputs and outputs up to time T, where $T > t$ is computed. The Kalman filter is used in the forward direction to compute the probability of X_t given $\{Y_{1,\dots,}Y_t\}$ and $\{U_1, \dots, U_t\}$. The backward recursions from T to t complete the computation and account for observations after time t. Finally, prediction is used to compute the probability of future states given observations up to time t.

Hidden Markov Models

Hidden Markov models also define probability distributions over sequences of observations $\{Y_t\}$. However, the output sequences are found by considering the observations at time t given a discrete hidden state S_t and the probability of transitioning to another hidden state (transition probabilities). Using the Markov property, the joint probability can be represented exactly the same as for state-space models by replacing X_t with S_t:

$$P(\{S, Y_t\}) = P(S_1)P(Y_1|S_1)\prod_{t=2}^{T} P(S_t|S_{t-1})P(Y_t|S_t) \qquad [1\text{-}17]$$

Furthermore the conditional independencies follow the same form as previously (see **Figure 1-23**). Any given state is represented by one of K discrete values, $S_t \in \{1, \dots, K\}$ and the transition probabilities $P(S_t|S_{t-1})$ are represented by a $K \times K$ transition matrix.

Two algorithms are often used to solve two particular inference problems. First, the recursive 'forward backward' algorithm is used to compute the posterior probabilities of the hidden states. The 'Viterbi' algorithm, also consisting of a forward and backward step, is used to compute the most likely sequence of hidden states. The 'Baum-Welch' algorithm can be used to learn the parameters of the model. The Baum-Welch algorithm makes use of the 'expectation-maximisation' (EM) algorithm and both a forward and backward step to find the maximum likelihood estimate for the parameters of the model given a set of observation sequences.

1.4.3 Copy Number Genotyping

Similar to the well-known and widely using methods for SNP genotyping [99-102] it is also possible to perform copy number genotyping. The goal here is to assign discrete copy number states across samples for any given genomic location. This is an essential step for performing accurate association studies on copy number variation. Because for any given genomic location across samples we do not have discrete values, rather a continuous distribution of log2 ratio values, the standard SNP genotyping methods cannot be used directly. Instead we need a way to confidently assign copy number state to individual samples, given the values observed across samples (see **Figure 1-21**). This is a relatively untapped field but is starting to receive more attention. One of the first publications describing an approach to address some of these problems modelled the quantitative CNV measurements using a Gaussian mixture model to assign copy number state [103].

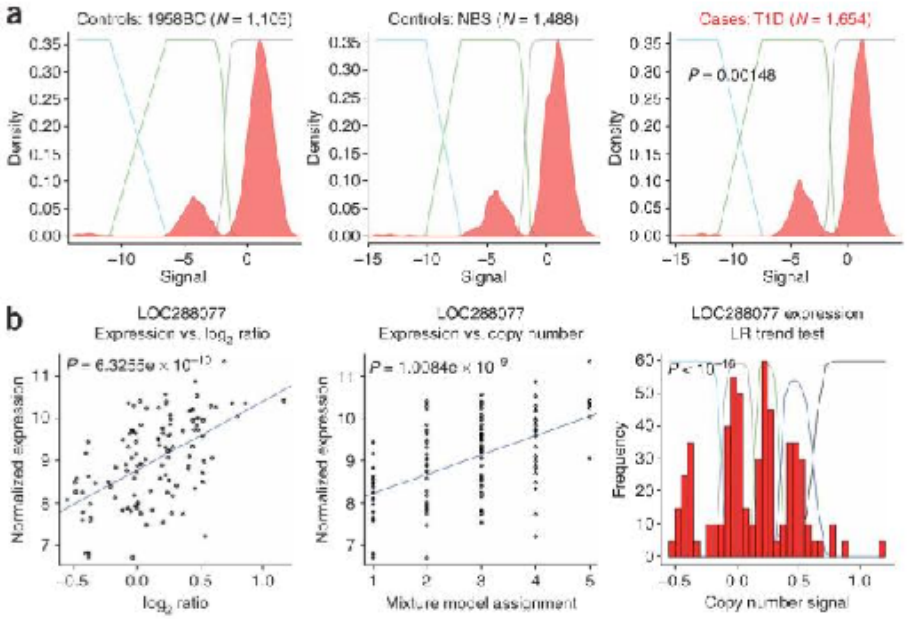


Figure 1-24 Examples of empirical CNV associations [103]. (a) The coloured lines reflect the posterior probability distribution for each mixture component in the fitted mixture model. (b) The first panel shows normalized gene expression against copy number measurement. The second panel shows normalized gene expression against mixture model assignment. The third panel shows a histogram of copy number measurement and the coloured lines represent the posterior probability distribution for each of the five copy number classes in the fitted mixture model used in the LR trend test.

1.4.4 Copy Number Tagging SNPs

An interesting question that has received some attention is to what extent copy number variants (CNVs) are in linkage disequilibrium (LD) with single nucleotide polymorphisms (SNPs). This is an important question that can address how much of the impact that CNVs have on common disease has been indirectly explored using GWAS studies. A paper describing such an analysis [104] found that most of the CNVs that they could type well were on average well

tagged by a SNP. In fact they observed that for these types of CNV, those with a minor allele frequency (MAF) greater than 10%, 79% had an r^2 greater than 0.8, indicating tagging almost to the same extend when SNPs are well tagged by other SNPs (see **Figure 1-22**).

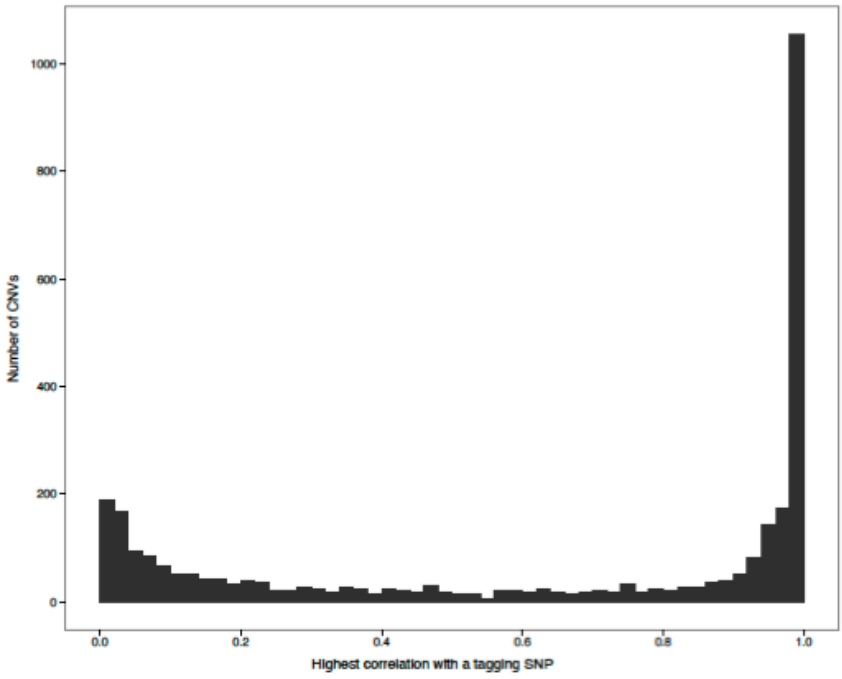


Figure 1-25 Histogram of maximum correlation r^2 between each CNV and a SNP within 1MB of the ends of that CNV [104].

This was an interesting observation and suggested that some CNVs could be intrinsically linked to certain SNPs. However, it is worth pointing out that although large-scale GWAS studies have been carried out on a number of different common diseases the role that CNV plays to these disease has in no way been fully explored, indirectly or otherwise.

1.4.5 Association Studies

An association study in the area of genomics can broadly be described as the search for genetic associations that occur more frequently than can be explained by chance. The associations that one would generally search for are genetic variants that are more prevalent in one group of individuals compared to another. These comparisons can take various forms; most commonly it would be between a set of diseased subjects against healthy controls [105] but it can be based on any number of other factors, for example ethic origin could be explored [106]. Genetic associations can be tested for between a phenotypic trait (such as eye colour) and a genetic marker (such as a SNP), or between two or more genetic markers and a phenotypic trait or even between the genetic markers themselves. The term linkage disequilibrium (LD) is used to describe the non-random association of two or more genetic markers. Thus genetic markers in strong LD with each other form haplotypes more or less frequently than can be expected by chance.

Case-control Design

A case-control design is the standard approach to genetic association testing. It compares cases that have been diagnosed with a particular trait against controls that do not display the trait [107]. A difference in the frequency of one or more genetic markers between cases and control can indicate that the presence of that genetic marker (or combination of markers) increases the likelihood of observing the specific trait. If an untested individual has the genetic marker (or combination) the chance that they will display the trait under study could be increased.

Trios

The Trio design for association testing is based on comparing family members against one another and searching for either genetic differences or similarities [108]. For example, consider the case where an offspring carries a particular disease but its parents do not. In this case one can direct the search towards *de novo* variants, those that are observed in the offspring but not in either parent. The justification for this is that since neither parent displays the trait (disease in this case) it is unlikely that the variant (or variants) responsible will be observed in their genomes. This is almost certainly an over simplification since certain traits will often be controlled by a number of different variants or even a specific combination of variants. The analysis of different combinations of variants with respect to different combinations of phenotypic traits is an exciting area of research and promises to deliver further insight into genetic associations. The Trio design can also be extended into more complex situations such as large family pedigree analysis [109].

1.5 COPY NUMBER MODES AND MECHANISMS

The number of copies of a gene (or a sequence of nucleotide bases) is the number of copies of that particular gene (or sequence of nucleotide bases) in the genome of an individual. It was historically thought that all genes in the human genome occurred in two copies however it is now known that for many genes the copy number can be only one (heterozygous deletion), more than two (gain) or even zero (homozygous deletion). This collective set of genetic variation is known as copy number variation (CNV) and can lead to gene dosage imbalances [110]. These changes in gene dosage influence most traits, contributing to what makes us all unique and includes the susceptibility and predisposition to various diseases [111-113]. Small or large sections of DNA can be gained or lost from any individual's genome and these differences in the copy number of genes or genic elements (e.g. exons and enhancers) can have a dramatic effect on an individual's health. Furthermore, since some CNVs effect gene dosage, resulting in differential gene product expression, they may play important roles in drug discovery and drug response prediction [114].

1.5.1 Gene Dosage

In some cases, for example the Bardet-Biedl syndrome type 5 associated BBS5 gene, both gene copies need to be perturbed for Bardet-Biedl syndrome [115] symptoms to be present. Genes displaying biallelic modes of action, such as BBS5, can be termed Haplosufficent in that a change in gene dosage is not sufficient to cause the associated phenotypes. In other words a single functional gene copy produces enough product to retain the wild-type condition, meaning that an abnormal or diseased state is not observed.

On the other hand some genes, for example the Cornelia de Lange syndrome type 1 [116] associated NIPBL gene, act under a monoallelic mode of action and can be termed haploinsufficient or dosage sensitive. For genes with monoallelic modes of action a change in gene dosage (normally a decrease) results in the associated phenotypes. Some genes can have both biallelic and monoallelic modes of action, for example the LRP2 gene causes Donnai-Barrow syndrome with a biallelic mode of action (both copies are perturbed) but has been associated with general Intellectual disability with a monoallelic mode of action (single copy loss).

1.5.2 X-linked and Hemizygous

In contrast to the autosomal chromosomes where both genders (male and female) have the same probability of inheriting a variant from their father or their mother, sex linkage is the phenotypic expression of an allele related to the chromosomal gender of an individual. The X chromosome in human males is termed hemizygous, meaning that it is present in only one copy. Hemizygosity can also be observed across the genome in males and females where one copy of a gene is deleted. X-linked inheritance can be dominant [117-119] or recessive [120-122]. Females affected with an X-linked dominant disorders have a 50% chance of passing the disorder on to an offspring of any gender. Sons of a male father with an X-linked dominant disorder will never inherit the disorder from their father whereas daughters will inherit the disorder 100% of the time. On the other hand females with X-linked recessive mutations are considered carriers and do not normally display symptoms (can rarely display very mild or different

symptoms). However all males with a X-linked recessive mutation will manifest with the disorder as they only have one copy of the X chromosome. Thereby male offspring of a female carrier will have a 50% change of inheriting the disorder and female offspring will have a 50% chance of inheriting the carrier state.

1.5.3 Digenic and Polygenic

In monogenic inheritance a single gene largely determines the presence of a phenotype or trait and mutations in one (dominant) or both (recessive) copies of the gene is sufficient for the trait to be expressed. For many of the yet to be understood genetic disorders it is likely that more complex situations such as dosage compensation, digenic or polygenic effects may hold the key [123, 124]. Digenic inheritance involves the interaction of two specific genes and mutations in at least one copy of both genes must be inherited for the associated phenotype to be expressed [125]. Polygenic inheritance involves phenotypes that are influenced by more than one gene. Polygenic traits often display a continuous distribution, such as weight, height, head circumference and eye colour. Polygenic traits that are influenced by environmental factor are termed multifactorial.

1.5.4 Mosaicism

Mosaicism is when two or more populations of cells with different genotypes exist in one individual. For example a specific form of Klinefelter syndrome (XY/XXY mosaic) can occur when some of the patient's cells contain XY chromosomes and some contain XXY chromosomes. Mosaicism can be present in the germline or somatic cells; somatic mosaicism is commonly caused by mitotic errors occurring during either mitotic recombination or somatic crossover, whereas germline mosaicism is a special form of mosaicism, where some but not all gametes carry a specific mutation. Certain genes can have a mosaic mode of action, such as the Proteus syndrome associated AKT1 gene [126].

1.6 THESIS OVERVIEW

In this thesis the development of novel analytical software for CNV detection and interpretation using a high-resolution array-CGH platform will be presented. As part of this research thesis all of the necessary analytical components, starting from raw array-CGH data, obtaining a set of potentially clinically relevant CNVs, including data normalisation, CNV detection, CNV annotation and CNV filtering are developed. A fully versioned and extensively tested analytical pipeline is established; encompassing all described methods for standard use in a large-scale genetic study based at the Wellcome Trust Sanger Institute (the DDD project). Finally the pipeline is applied to a set of over 1000 normal controls and results are presented focused on both the analytical validity of the methods and the scale and type of CNVs found in apparently healthy individuals.

Chapter 2 describes the full technical details, thorough assessment and improvement of four major analytical components. First the development and optimisation of a change point detection package, CNsolidate, is described, which makes use of multiple weighted change point detection algorithms and includes an adaptive learning algorithm for estimating individual algorithm weights given certain characteristics of the input data. Second a novel Bayesian framework for allowing array-CGH sample tracking using copy number tagging SNPs is presented. A description of how the individual probabilities for the model are generated and updated followed by an assessment into the performance of the overall sample tracking values in terms of sample mismatches and discriminatory power is provided. Third, details of generating a combined common CNV reference set (CNV consensus) are presented, using a number of high quality studies with different sample sizes and spanning the genomic size range for common CNVs. Details of the effect that adding merged CNV events (CNVEs) called by CNsolidate has on the overall number, type and frequency of CNVEs within the CNV consensus are given. A general approach for classifying CNVs as either novel, rare or common based on positional similarity, CNV type states and population frequency estimates is described. Finally, details of a rule based approach to flagging CNVs of potential clinical significance are described and the results of filtering DDD patient CNV data in terms of the overall characteristics of flagged CNVs are provided. The discovery of an analytical complication based on the clinical filtering results and an association based analysis to search for problematic array-CGH data points (probes) is performed.

Chapter 3 describes the implementation of a large-scale analytical pipeline encompassing all the analytical methods developed in **Chapter 2**. Some general best practice approaches to generating a maintainable and robust code base using version control are described. Examples of using a large compute farm and making use of a load sharing facility (LSF) to increase pipeline throughput are shown. Finally details of the approach chosen for organising and maintaining pipeline data and results (inputs and outputs) using a structured file system and database are described.

Chapter 4 is made up of a manuscript currently under review in Genome Research and presents a detailed assessment of the performance for CNV discovery achieved by CNsolidate. It includes results and performance assessments of another algorithm (VICAR) which was developed by a colleague and which I incorporated into the DDD CNV analytical pipeline. The manuscript also includes a refinement of the CNV mutation rate by defining a set of high confidence *de novo* CNVs with a median resolution of 15.2 Kb. Finally suggestive evidence for a paternal bias in the formation mechanisms of *de novo* CNVs is presented by performing a parent of origin analysis.

Chapter 5 is made up of a drafted manuscript and presents results from applying the clinical filtering pipeline for CNVs to over 1000 normal control individuals. We show the overall number of flagged variants and report a surprisingly high potential feedback rate for CNVs detected in apparently healthy individuals. When looking in detail at the individual characteristics of flagged CNVs we present a small number of CNVs in genes relating to relatively common rare disorders or in genes thought to be associated with complex rare disorders such as autism and epilepsy. Additionally we discover two female control samples with CNVs in the X-linked recessive gene STS and who are therefore predicted to be carriers of X-linked Ichthyosis (XLI) disorder.

Concluding remarks and future direction are detailed in ***Chapter 6***

2 | Copy Number Discovery and Interpretation

2.1 INTRODUCTION

Individual change point detection algorithms often display variable performances [20, 127-129]. The definition of an optimal set of parameters for achieving a certain level of performance is rarely straightforward, especially where data qualities vary. We propose a combined change point detection package, CNsolidate, which makes use of an expert voting system. Using this approach, it is possible to rank detections based on differential weighting functions between component algorithms (voters).

CNsolidate derives all of its parameter definitions and weighting functions using a number of predictive variables drawn from the input data characteristics. Accurately ranking detections across data sets of variable qualities allows the use of a single threshold to control the balance between type 1 and type 2 error rates. CNsolidate has been primarily designed for direct use in the detection of copy number variable genomic regions for the Deciphering Developmental Disorders (DDD) project based at the Wellcome Trust Sanger Institute (WTSI) [29].

A frequently used approach to filtering genomic variation data, obtaining a set of potentially interesting variants, is to use a control set of variation to compare the given test set against. Indeed, for the filtering of single nucleotide polymorphisms in GWAS studies the database dbSNP is often used to exclude variants that have been previously observed at a given frequency within the general population [130]. For copy number variation there are a number of studies that have been performed to search for common CNV within the general population [13, 50]. Furthermore, databases such as dbVar [131] are now attempting to collate information across multiple studies of common CNV to provide a single resource for variant filtering.

There are a number of problems when using generic genomic variation reference sets to filter variants against [132-134]. Different studies of genomic variation will have been performed using different sample sizes, will have utilised technologies of different resolutions and change detection will have been performed using algorithms displaying different sensitivities and specificities. Furthermore, data quality control and experimental processing will be different across multiple studies. The combination of all these factors can introduce a number of biases into any combined reference set resulting in a potentially inaccurate filtering process. Due to the mentioned difficulties, we elected to create a CNV consensus reference set incorporating a number of high quality studies. This section describes the integration of several control data sets into the single CNV consensus reference set. More generally, it gives an overview summary of the component data sets that make up the current CNV consensus set. The CNV consensus set has been created for direct use in determining potentially pathogenic variants for patient data generated as part of the DDD project based at the WTSI. The CNV consensus reference set is publically available for download from the DECIHPER database [16, 135-137].

One very important aspect of the DDD project is that of accurate sample and data tracking throughout the various pipelines. This is critical to the overall success of the project since the data sets produced could potentially be used for direct clinical diagnosis via the regional NHS genetic services within the UK and Ireland. Although each variant reported will ultimately require validation in an accredited laboratory (regional genetic services), the DDD project takes sample and data tracking extremely seriously. A number of the existing WTSI data generation pipelines track their samples and data via a molecular barcode in the form of 30 SNPs typed using a Sequenom assay [138]. The moment that a DNA sample enters the sample storage and extraction facilities (sample logistics) it has SNPs typed using the Sequenom platform. These SNP genotypes are stored in a large oracle database and can be used as a final check once any further data has been generated. If the SNP genotypes produced as a result of the sequencing or genotyping pipelines do not match those typed on the Sequenom assay for a given sample this indicates a sample / data tracking failure.

For the Sanger genotyping and sequencing pipelines this lookup is trivial as the data types generated are the same as the Sequenom assay, specifically SNP genotypes. Thereby the check for sample / data concordance is simply the number of matched genotypes between the assays under question. Failures are defined using a threshold on the number of genotype 'mismatches'. For the high throughput array-CGH laboratory and analytical pipelines there was no sample / data tracking facility available as standard at the WTSI. To maintain sample level information throughout the laboratory procedures the DDD informatics team has developed a laboratory information management system (LIMS) using a Java based (spring) framework. For the data tracking throughout the array-CGH analytical pipelines we have developed a method that makes use of copy number tagging SNPs to allow the array-CGH (continuous log2 ratio) data to be correlated with SNP genotypes obtained via the Sequenom assay run during sample reception.

The end result in terms of the service provided by the DDD project to the NHS genetic services is the feedback of potentially clinically relevant variants via the DECIPHER database. To this goal the DDD project has developed a number of variant filtering approaches including a rule-based pipeline for prioritising CNVs of potential clinical interest. This pipeline incorporates information about CNVs in genes that have previously been associated with genomic disorders and additionally predictions of CNVs seen at low enough frequency in apparently healthy individuals to be of general clinical interest. The CNV filtering pipeline has been designed to allow iterative reporting rounds maintaining information across multiple clinical filtering versions. This allows the project to be initially cautious when predicting the clinical relevance of CNVs, refining the filtering rules as more data is obtained and as the understanding of the causes of genomic disorders is improved.

2.2 METHODS

2.2.1 CNsolidate

Array Platforms

The DDD project utilises two different array technologies in the form of array-CGH and SNP genotyping platforms. The array-CGH array is composed of two 1Million probe Agilent arrays and has been heavily targeted towards genes and ultra-conserved elements throughout the human genome. The entire set of GENCODE exons [139], along with regulatory and mRNA coding elements have been tiled, using a minimum of five oligo-nucleotide probes, on the Agilent array-CGH arrays. The rest of the array content is spent on ensuring the presence of an ultra high-resolution backbone with a median probe spacing of 5Kb.

The SNP genotyping array is a customised version of the Illumina Omni-one quad chip (SangerDDD_OmniExPlusv1_15019773_A). Extra content has been added to standardise the coverage of the array using a ”largest first” gap filling procedure. The gap filling process is aimed at targeting the largest gaps in array coverage first and additionally inserting the best quality probe within the ”central gap region” before moving to the second largest gap in array coverage.

The two million probe Agilent array-CGH array is used as the discovery platform due to both its higher density and improved sensitivity. All the results described for CNsolidate are derived from data sets generated using the Agilent 2 x 1M custom array-CGH microarray (Agilent; Amadid No.s 031220/031221).

Component Algorithms

CNsolidate uses the combination of 12 independent change point detection algorithms to detect data segments that are potentially different to the background within a time-series. These algorithms's encompass both published and novel methods and have been tuned in concert to achieve high sensitivity while maintaining an acceptable level of specificity across a large range of data qualities.

By using a combination of algorithms to detect change points from time-series data it is possible to rank segments based on some prior knowledge. This can be done according to both the particular combination of algorithms that detected a segment ('naive voting') and by estimating each algorithms performance given specific predictive variables that can be measured from the input data characteristic ('expert voting').

In this section the 12 different change point detection algorithms currently included in CNsolidate are listed and a brief description of their methodologies is given. We reference previously published algorithms accordingly and denote novel change point detection algorithms using a * in their title declarations.

GADA - Genome Alteration Detection Analysis

First, the GADA algorithm [140] describes copy number across a time series using piece-wise constant (PWC) vectors. The underlying mean hybridization intensity X_m is PWC since it depends only on the number of DNA copies:

$$y_m = x_m + \varepsilon_m$$

where y_m represents the log2 of the relative hybridization intensity observed by probe m; x_m represents change in hybridization intensity due to altered copy number, ε_m is a zero-mean array noise.

Any PWC vector x with K breakpoints can be represented by a linear combination of K step vectors f_i plus a constant vector f_0. An empirical Bayes approach, SBL, is applied to infer locations of change indicative of copy number alteration. The SBL approach uses a maximum a posterior (MAP) estimate, where the observation model $p(y|w)$ specifies the goodness of fit measure and the prior distribution for the weights $p(w)$ specifies the sparseness measure.

A suboptimal backward elimination (BE) procedure is used to rank the inferred breakpoints. Breakpoints with lower statistical evidence t_j are recursively eliminated given the chosen significance parameter T.

*ADM3 - Automated Detection Algorithm 3

ADM3 is an interpretation of the ADM2 algorithm, originally developed by Agilent Technologies and contained in the Genomic Workbench software.

The ADM interval score is calculated using a vector of pairs $(l_1, le_1), (l_2, le_2), \ldots, (l_n, le_n)$ where l_i is the log-ratio signal for the i^{th} probe and le_i is the log-ratio error for the i^{th} probe.

Define $q_i = 1/(le_i)^2$, the ADM score S for interval I is defined as:

$$S(I) = \frac{\sum q_i l_i}{\sqrt{\sum q_i}} \qquad [2\text{-}1]$$

A noise term T, the derivative log ratio spread, and a scaling constant F are applied to transform the ADM score relative to data quality.

CBS - Circular Binary Segmentation

CBS is a modification of binary segmentation first developed by [141]. The version used is based around the 'dna.copy' bioconductor package. The modifications made are aimed towards increased speed performance and improved consistency across varying noise backgrounds. Let $X_1, X_2, \ldots, X_n$ be a sequence of random variables. An index v is called a change point if $X_i, \ldots, X_v$ have a common distribution function F_0 and $X_{j>v}$ has a different common distribution function F_1 until the next change-point.

In CBS, where the data are normally distributed with a known variance, the likelihood statistic for testing the null hypothesis is given by $Z_B = max_i \leq_{i<n} |Z_i|$

where,

$$Z_i = \{1/i + 1/(n-i)\}^{-\frac{1}{2}}\{S_i/i - (S_n - S_i)/(n-1)\} \qquad [2\text{-}2]$$

The null hypothesis of no change is rejected if the statistic exceeds the upper α^{th} percentile of the null distribution of Z and the location of the change-point is estimated to be i such that $Z = |Z_i|$.

SWArray - Dynamic Programming Algorithm

For the SWArray algorithm a modified version of the 'tilingArray' bioconductor package [142] was used. They developed an implementation of dynamic programming, combining a segmentation model with a mixture model. Their method combined the dynamic programming algorithm (DP) and expectation maximization (EM). This hybrid algorithm, called dynamic programming expectation maximization (DP-EM) estimates the parameters of the model by maximum likelihood.

Due to the computationally intensive nature of a dynamic programming approach, the method has been modified to use low-cost matrices, thereby reducing the search space. CNsolidate pre-segments the data into manageable data 'chunks' prior to applying the DP-EM algorithm. These segments are offset and summed three times.

**CNCP - Copy Number Change Points*

CNCP is based on the CUSUM (or cumulative sum control chart) approach to change detection originally developed by E. S. Page.

In CNCP the cumulative sums are only allowed to extend across data measurements by defining a certain set of criterion and are reset once a change has been observed. The likelihood function ω is based on the distribution of a number of data points preceding the current point in the control chart.

Let X equal the data measurements, ω equal the likelihood function, N equal the number of data measurements and S equal the cumulative sum.

First we calculate the mean of X:

$$\bar{X} = \frac{(X_1 + X_2 +, \dots, + X_N)}{N} \quad [2\text{-}3]$$

Then start by setting $S_0 = 0$ and $\omega_0 = 0$,

$$S_{i+1} = \left(S_i + (X - \bar{X})\right) - \omega_i \quad [2\text{-}4]$$

with i from 1, 2, ..,N.

As more values are added to the cumulative sum, if the majority of measurements were positive with respect to the mean and likelihood function the sum will steadily increase. When the value of S exceeds a certain threshold a change has been found.

Intuitively, a segment of the CUSUM chart that displays an upward slope indicates a period of time where the measurements tended to be above the mean given the likelihood function.

FastCall - Fast Change Point Detection

The FastCall [143] method is based on a mixture of truncated normal distributions. It classifies data segments into one of four states, loss, neutral, gain and multiple gain. FastCall requires as input 'pre-detected' data segments and then models the mean level of a segment as a mixture of five truncated normal distributions.

Due to the fast execution properties of FastCall We were able to use a greedy segmentation algorithm. Let, X equal the raw ratio values, $\bar{X}$ equal the mean of the current test segment, F_c equal the FastCall algorithm, s equal the FastCall state and N equal the length of X.

$$s_i = F_c(\bar{X}) \quad [2\text{-}5]$$

with i and j from $1, 2, \ldots, N-1$. The current test segment $\bar{X}$ is the mean of the raw ratio values, X from $i \ldots j$. When the current state classification, s_i is equal to the previous classification, $j = j + 1$ and the process continues. When the current state classification s_i is not equal to the previous classification, $i = j$ and $j = i + 1$. This process operates on the raw ratio values from each chromosome separately and defines a state classification vector of the same length for each. These state classification vectors are then used to dictate the starting and ending positions, and the mean of outlier segments across the data measurements where consecutive state classifications are the same but not the normal state (copy number 2).

*STFE - Simple Threshold Feature Estimation

STFE uses a noise dependent threshold and a self-adjusting extension rule to estimate variable segments within time-series like data measurements.

We define T, the detection threshold, as:

$$T = nF \times Dv \quad [2\text{-}6]$$

where, nF is a threshold scaling factor and Dv is the 68^{th} percentile of the absolute, median normalised data measurements.

The initial extension parameter, $E = 1$.

Let $X_1, X_2, \ldots, X_n$ be the ordered data measurements, when any $X_i > T$, the detection vector dI is defined as:

$$dI_i > T, X_i, .., X_{i+E} \quad [2\text{-}7]$$

As the length of dI increases E is adjusted by:

$$E = \sqrt{L(1 - \left(\frac{c}{L}\right))} \quad [2\text{-}8]$$

where, L is the length of the detection vector and c is the number of data values $X_i < T$ inside the detection vector. Once $\sum dI \geq T \geq E$, extension is terminated, $E = 1$ and $i = (i - E) + 1$.

SMAP - Segmental Maximum A Posteriori Approach

The segmental maximum a posteriori approach to genome-wide copy number profiling - SMAP [144], used is based on a discrete-index Hidden Markov Model and incorporates genomic distance and overlapping probes. They use a six-state model in place of the conventional three-state model. The version used is a modification of the available bioconductor package.

Their HMM is a pair $\hbar = (S, \lambda)$, where $S = \{S_i\}_{i=1}^{N}$ is a set of N copy number states, such that $q_t \in S(1 \leq t \leq T)$, and $\lambda(\sqcap, \wedge, \Omega)$ are parameters for the model.

The probability of starting in copy number state $s_i(1iN)$ for data point one is specified by the initial probabilities, $\Pi = \{\pi\}_{i=1}^{N}$. Each pair of copy number states is connected by HMM specific transition probabilities $A = \{a_{ij}\}_{i,j=1}^{N}$ that specify the probabilities of transition between states s_i and $s_j(1i, jN)$ between any two consecutive data points in the sequence.

SMAP suggests a six state model by default but is not restricted to six states. In CNsolidate SMAP is used with only three states due to observed complications in detection performance when using the six state model across data of variable qualities.

*Vwalk- Variance Spike Walking

Vwalk makes use of self-adjusting sliding windows to search for spikes in variance, characteristic of change-point locations, across time-series like data.

We define the detection threshold vT as:

$$vT = var(D)sf \qquad [2\text{-}9]$$

where *sf* is a scaling factor and *D* are the data measurements.

Using an initial window size *w* the data measurements are tested and where $var(X_i, \dots, X_{i+w}) > vT$ a potential segment starting position has been found.

Next the window size *w* is incremented so long as $var(X_i, \dots, X_{i+w}) < E$, where E is $var(X_{i-1}, X_w)/2$. This continues until $var(X_i, \dots, X_{i+w}) > E$ at which point the potential segment ending position has been found, $i = w$ and *w* equal the initial window size.

We apply this process twice across the data measurements for each chromosome separately using two different window sizes. The first window size is small (4 data points) and is aimed at increasing the sensitivity of outlier detection from the method, however this window size tends to over segment and result is multiple small detections. Thereby a second larger window size (20 data points) is used and a final process, using fixed cut-offs on mean ratio differences, is used to merge adjacent segments where appropriate.

*ViteRbi

ViteRbi is a simple three state Hidden Markov Model using the standard Viterbi algorithm to compute the likely sequence of hidden states ('Viterbi path').

Given the model:

$$\delta_t(i) = \max P(q(1), q(2), \dots, q(t-1); o(1), o(2), \dots, o(t) | q(t) = q_i) \qquad [2\text{-}10]$$

where $\delta_t(i)$ is the maximal probability of states of length *t* and end in state *i*. The Viterbi algorithm uses maximisation at the recursion and termination steps, and keeps track of arguments that maximize $\delta_t(i)$ by storing them in the N by T matrix ϕ.

The optimal state sequence is retrieved at the backward step:

$$q_t = \phi_{t+1}(q_{t+1}), t = T-1, T-2, \dots, 1 \qquad [2\text{-}11]$$

Finally, segments of the optimal state sequence vector q displaying a non-normal state are split into discrete distance intervals.

*SMUG - Stochastic Model Under 'Gain'

SMUG is a non-stationary Hidden Markov Model using a time-inhomogeneous Markov chain.

In other words, as the time spent in a state increases; the transition probabilities alter given a prior expectation on the likely time to be spent in that particular state.

The Markov property remains unchanged. We can state that P depends on i, j and n, and call it $p_{n,ij}$:

$$P(X_{n+1} = j | X = i, \hbar_n) \qquad [2\text{-}12]$$

These can then be assembled into a transition matrix P_n.

The Chapman-Kolmogorov Equations show that the m-step transition probability $P(X_{n+m} = j | X_n = i)$ is the $(i,j)^{th}$ element of the matrix obtained via matrix multiplication. The behaviour of the population as a whole can be treated as time-homogenous because of statistical equilibrium.

BCP - Bayesian Change Points

CNsolidates version of BCP is based on an R package [145] implementing a Bayesian approach to change point detection from Barry and Hartigan 1993.

In the original implementation by Barry and Hartigan, they assumed that the probability of a change point at a position i is p, independently at each i.

In their Bayes procedure, under this assumption, $N(\mu_i, \sigma^2)$, the prior distribution μ_{ij} is choosen as $N(\mu_0, \sigma)_0^2/(j-i)$ and the calculations are $O(n^3)$. In the Erdman and Emerson R implementation they use a MCMC approximation of the prior distribution that is $O(n^2)$.

The algorithm uses a partition $p = (U_1, U_2, \ldots, U_n)$, where $U_i = 1$ indicates a change point at position $i+1$. In each step of the Markov chain, at each position i, a value of U_i is drawn from the conditional distribution of U_i given the data and the current partition. The transition probability, p, for the conditional probability of a change point at the position $i+1$, can be obtained from the simplified ratio presented in Barry and Hartigan:

$$\frac{p_i}{1-p_i} = \frac{P(U_i = 1 | X, U_j, j \neq i)}{P(U_i = 0 | X, U_j, j \neq i)} = \frac{\left[\int_0^\gamma p^b (1-p)^{n-b-1} dp\right]\left[\int_0^\lambda \frac{w^{b/2}}{(W_1 + B_1 w)^{(n-1)/2}} dw\right]}{\left[\int_0^\gamma p^{b-1} (1-p)^{n-b} dp\right]\left[\int_0^\lambda \frac{w^{b-1/2}}{(W_0 + B_0 w)^{(n-1)/2}} dw\right]} \qquad [2\text{-}10]$$

where, W_0, B_0, W_1 and B_1 are the within and between block sums of squares obtained when $U_i = 0$ and $U_i = 1$ respectively, and X is the data. γ and λ are tuning parameters with values between 0 and 1.

BCP is both the most complex and the least reliable algorithm within CNsolidate. Currently it is not used under the default settings, as its successful completion cannot be guaranteed. In future releases of CNsolidate we hope to improve the performance and reliability of the BCP algorithm.

Algorithm Combination

Each individual algorithm results in the identification of a number of potentially significant segments throughout the genome. These segments contain a start and a stop position with a number of internal data points, they denote a region of the time-series data that is potentially significantly different to the background noise.

CNsolidate makes use transitive set theory to build up well-ordered sets of ordinals for each chromosome separately. Each of these ordinals is itself a well-ordered set of detected segments.

Constraints:

We define the binary relationship R as,

$$R = (C_i(S_k) \leq C_i(E_{k+1}) \;\&\; C_i(E_k) \geq C_i(S_{k+1})) \qquad [2\text{-}14]$$

where, C_i equal a well-ordered set of segments from chromosome i, S_k equal the start position and E_k equal the stop position of segment k from set C_i

Transitive Clustering

We use the binary relationship R to transitively cluster features to build up the ordinals (well-ordered sets of detected segments).

Given an initial ordinal OR, a subset of segments from the well-ordered chromosomal set C, the ordinals are defined by:

$$\forall_x \in OR: \forall_y \in C: if\ R(OR_x, C_y)\ then\ \cup\ OR, C_y \qquad [2\text{-}15]$$

Iterative Closure

Full iterative closure is achieved once the set C can no longer provide member segments to any of its ordinals ORS.

In other words,

$$\forall OR \in ORS: \forall_x \in OR: \forall_y \in C: \neq R(OR_x, C_y) \qquad [2\text{-}16]$$

The result of this iterative process is a set of ordinals for each chromosome. Each ordinal is itself a well-ordered set of detected segments. Furthermore, each ordinal contains information about which combination of algorithms and in which context each of its member segments were detected.

Feature Definition

To define non-overlapping, unique features from the well-ordered chromosomal sets we apply a method based on the meeting combination and the meeting arrangement observed in each of the ordinals.

The input to this method is two matrices for each ordinal,

$$MC_{m,n} = \begin{pmatrix} mc_{1,1} & \cdots & mc_{1,n} \\ \vdots & \ddots & \vdots \\ mc_{m,1} & \cdots & mc_{m,n} \end{pmatrix} \qquad [2\text{-}17]$$

And,

$$MA_{m,n} = \begin{pmatrix} ma_{1,1} & \cdots & ma_{1,n} \\ \vdots & \ddots & \vdots \\ ma_{m,1} & \cdots & ma_{m,n} \end{pmatrix} \quad [2\text{-}18]$$

Where, m is the internal breakpoint index, and n is the algorithm index across the ordinal. Thereby, $mc_{1,1}$ equal the meeting combination score and $ma_{1,1}$ equal the meeting arrangement score for algorithm one at breakpoint one respectively.

We then define the weighted ordinal scoring vector ϕ, as:

$$\phi = \frac{\sum_{i=0}^{n} mc_{i,n}}{\sum_{i=0}^{n} ma_{i,n}} \quad [2\text{-}19]$$

The ϕ values reflect the relative difference between the number of algorithms in agreement at a potential breakpoint location and the number of algorithms that show extension across a breakpoint. When ϕ_i is greater than 1, there is more evidence for a breakpoint than an extension and the ordinal is split. By default, we do not apply any weighting to the individual algorithms during this feature definition step. Meaning that, it is only the number of algorithms, not the type that is considered for the ϕ values. However it is possible to incorporate some prior knowledge, for example, if a particular algorithm is known to over-segment across regions its *mc* score could be down-weighted. Conversely, if an algorithm is known to over-extend across multiple breakpoints, its *ma* score could be down-weighted.

Segment Exclusion and Breakpoint Mapping

To exclude poor quality segments there are two hard constraints. The minimum absolute mean ratio of a segment, *mRS* and the minimum number of data points needed to define a segment *mDP*. All ordinals must obey these constraints to be placed into the filtered ordinal set *FOR*.

$$\forall_x \in OR: |OR_x(mR)| \geq mRS \;\&\; OR_x(mD) \geq mDP : \cup FOR, OR_x \quad [2\text{-}20]$$

where, *mR* and *mD* equal the absolute mean ratio and the number of data points from the maximum union of each ordinal respectively. For breakpoint mapping, there is a single soft constraint, *brkM*, which defines the minimum absolute ratio value that is allowed at both ends of the ordinal. It is related to the mean ratio value of the ordinal, *orM,* or a defined threshold, *brkT*.

$$brkM = \max(brkT, (orM/2)) \quad [2\text{-}21]$$

If data points at either end of the ordinal are less than *brkM*, extension outside of the ordinal is attempted. If this fails, data points inside the ordinal are tested until the end values are greater than *brkM*.

Segment Merging

Finally, adjacent ordinals can be iteratively merged into a single ordinal so long as two constraints hold true.

Let OR_r be the mean ratio of the ordinal and OR_s be the number of data points in the ordinal, and let *rp* and *sp* be adjustment parameters.

Define

$$R = OR_r \times rp \quad [2\text{-}22]$$

and

$$S = \sqrt{OR_s} \times sp \quad [2\text{-}23]$$

Let ORN be the nearest neighbour to OR and let dr be the absolute difference in mean ratios and ds be the number of data points between OR and ORN respectively.

$$if\ dr \leq R\ \&\ ds\ \leq S\ then\ \cup OR, ORN \quad [2\text{-}24]$$

This process continues, across all ordinals ORS, selecting the nearest neighbour and testing R and S until full closure is achieved:

$$\forall_x \in ORS: (OR|ORN)\ dr_x > R_x > ds_x > S_x \quad [2\text{-}25]$$

This is the final step in the definition of the locations of potentially significant change point intervals across the time-series. It results in a well-ordered set of ordinals, each of which is itself a well-ordered set.

Weighting Functions

To make the best use of all these results and to place a more reliable level of significance on each detected feature, we have created a score assignment method. This method makes use of a number of predictive variables and the estimated performance of each algorithm ('voter') across a range of data qualities. As a result it is possible to rank features according to some prior knowledge.

Currently this knowledge is based on:

1. Individual algorithm performances given three different measures of data noise characteristics
2. The importance of three specific predictive variables on detection performance in general

Noise Dependent Algorithm Weights

To estimate the performance-based weights for each algorithm we test the detection accuracy of each algorithm using three different estimates of data noise levels. For each algorithm the false alarm rate was tested using over one million data simulations (see **Appendix**).

First we use the *rp68*, the 68th percentile of the absolute median normalised log2 ratio values as an estimate of the amount Gaussian 'white' noise present in the data set.

Second the *dLRs*, the derivative log ratio spread:

$$dLRs = \frac{IQR(d)}{4(erfinv(0.5))} \quad [2\text{-}26]$$

Where, *d* is an vector of differences between log2 ratios of adjacent probes, *erfinv* is the Inverse Error Function and *IQR* is Inter Quartile Range.

Third the *dydLRs*, or 'wave score':

$$dyDLR_i = dLRs(dspan) \quad [2\text{-}27]$$

with *i* from 1, 2, ..,*N.*

where *dspan* is a vector of log ratio values with an equally spaced index *i* and *N* is the maximum index spacing.

$$dydLRs = \sum |dDS| \quad [2\text{-}28]$$

where *dDS* is an array of differences between the *dyDLR* values. This value gives an estimate of the scale of auto-correlation present in the data values.

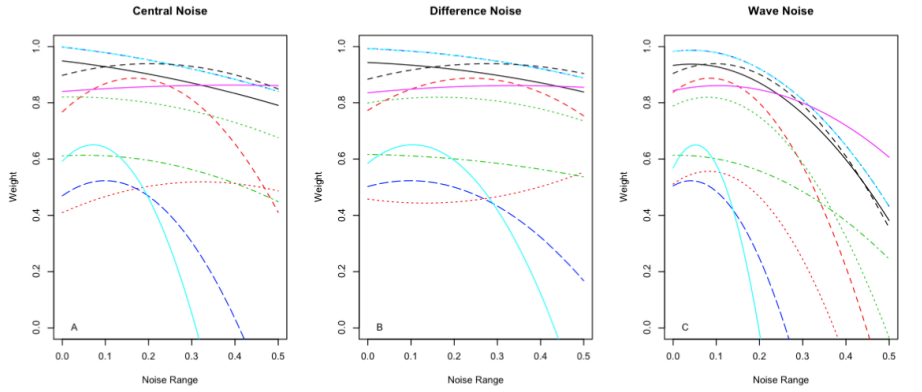


Figure 2-1 Noise dependent weight functions for 11 algorithms. Left - Central noise measure vs. Weight. Middle - Difference based noise vs. Weight. Right - Wave based noise measure vs. Weight.

Above (see **Figure 2-1**) the derived noise dependent weighting functions for 11 of CNsolidates algorithms is shown. BCP is excluded due to the fact that its successful completion cannot be guaranteed, as previously mentioned. Each of the noise values (central, difference and wave) has a different scale and has been rescaled onto a 0-0.5 range for the purpose of display. These curves were generated by assessing the relationship between the noise ranges and the Type I and Type II error rates via a large number of data simulations (see **Appendix**). During these simulations we varied the three noise predictors and assessed the performance of each algorithm separately.

Overall each algorithm displays a different relationship between its estimated performance and the various different data noise types and levels. Although a

few algorithms show a general poor performance, being very sensitive to differences in noise levels, the majority of algorithms show reasonable performance across a large portion of the noise ranges.

Interestingly most algorithms show the expected decrease in performance when noise levels increase, however a few display the inverse relationship to noise levels. This is useful as its means that at most combinations of noise types and levels there are a number of high performance ('expert') algorithms. The wave based noise estimation has the largest effect on algorithm performance across the board, with all algorithms displaying a decrease in performance as wave levels increase.

Feature Dependent Algorithm Weights

Next, we estimate the false alarm rate given some predictive values that can be calculated from each detected segment. Again we estimate the relationship between each of these predictive variables and the false alarm rate via a large number of data simulations (see **Appendix**).

Currently three predictive variables are used during this weighting procedure. The absolute mean ratio of a segment, the number of data points in a segment and the variance of the data points across the segment.

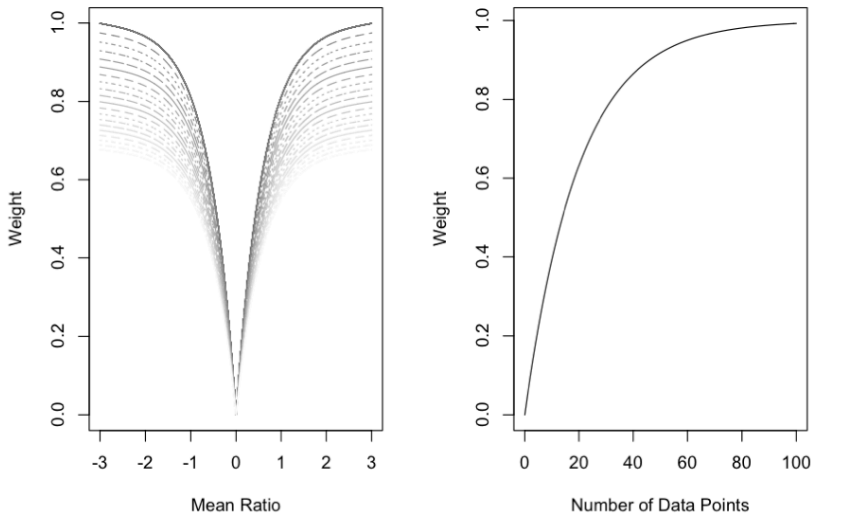


Figure 2-2 Feature Dependent Weighting Functions; Left - Ratio dependent weights. Right - Number of data point's dependent weights.

Above (see **Figure 2-2**) the estimated weighting functions for the mean ratio of a segment and the number of data points in a segment respectively are shown. The ratio weighting function has an additional noise term, where increased levels of noise result in the weight being reduced across the range. In the left hand panel, the shade of the lines relates to a difference in noise, with noise increasing as the shades becomes lighter.

In the right hand panel, the size weighting function, the curve plateaus at around 100 data points. Once a change point interval has 100 or more data points it was

detected 100% of the time. This is perhaps not too surprising but it does result in an added complication. A segment containing 100 data points will receive approximately the same size dependent weight component as one containing 10,000 data points using this weighting function.

Detection Scores

Weighted Confidence Score

For each detection, we assign a weighted confidence score given the predictive values mentioned above. We define the weighted confidence score *w* as:

$$w = \left(\frac{(rW + sW + vW)}{3} + \frac{\sum_{k=1}^{n} (cV_k, dV_k, wV_k)}{N} \right) / 2 \qquad [2\text{-}29]$$

Where, fW, sW and vW equal the feature dependent weights for the mean ratio, size and variance respectively, cV , dV , wV equal the noise dependent weighted scores for the rp68, the dLRs and the dydLRs for algorithm k respectively, and where, N equals the total number of algorithms used.

The weighted confidence score (w) is thus a composite value made up the noise dependent and feature dependent weighted scores.

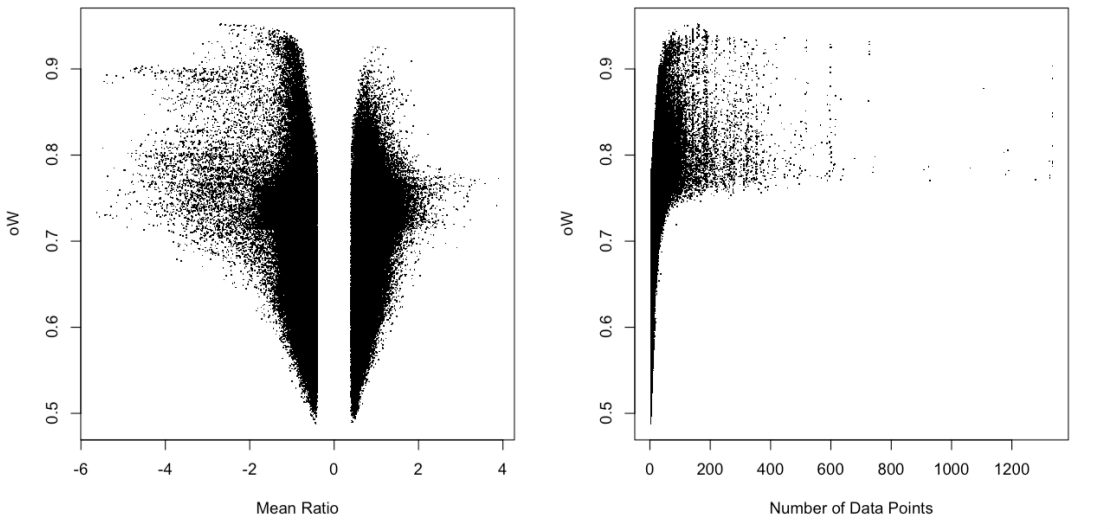


Figure 2-3 Relationship between weighted score and two predictive variables. Left - Weighted score vs. mean ratio. Right - Weighted Score vs. number of data points.

The above plots (see **Figure 2-3**) show the relationship between the wscore and the two predictive variables mean log2 ratio (left) and number of data points (right) for a number of CNV detections. Although there is a general correlation

observable, as expected it is not perfect since the wscore is a composite value made up of multiple predictive values.

Local Significance Value

Additionally, for each detection, we assign a p-value calculated via the two sample Welch's t-test. The test is between the population mean of data points inside $\bar{X}_1$ and those directly outside, but not inside another, detected segment $\bar{X}_2$.

The number of data points drawn from directly outside the segment (on both sides) to make up the second distribution (with the population mean $\bar{X}_2$) is controlled by a factor f. Thereby the number of data points to draw from both sides of the segment dn is defined as, $dn = sn/f$ where sn equal the number of data points inside the segment.

By default we set $f = 2$ in CNsolidate meaning that half the number of data points within the segment are drawn from each side. These are then combined into a single distribution and compared to the distribution of data points inside the segment via the t-test:

$$t = \frac{\bar{X}_1 - \bar{X}_2}{s_{\bar{X}_1 - \bar{X}_2}} \qquad [2\text{-}30]$$

where

$$s_{\bar{X}_1 - \bar{X}_2} = \sqrt{\frac{s_1^2}{n_1} + \frac{s_s^s}{n_2}} \qquad [2\text{-}31]$$

and s^2 is the unbiased estimator of the variance of the two samples, n_1 and n_2 equal the number data points in distribution 1 and 2 respectively and the degrees of freedom are calculated using the Welch–Satterthwaite equation:

$$d.f. = \frac{(s_1^2/n_1 + s_2^2/n_2)^2}{(s_1^2/n_1)^2/(n_1 - 1) + (s_2^2/n_2)^2/(n_2 - 1)} \qquad [2\text{-}32]$$

Finally we use the Bonferroni correction for multiple testing where the p-values are multiplied by the total number of tests (CNV detections).

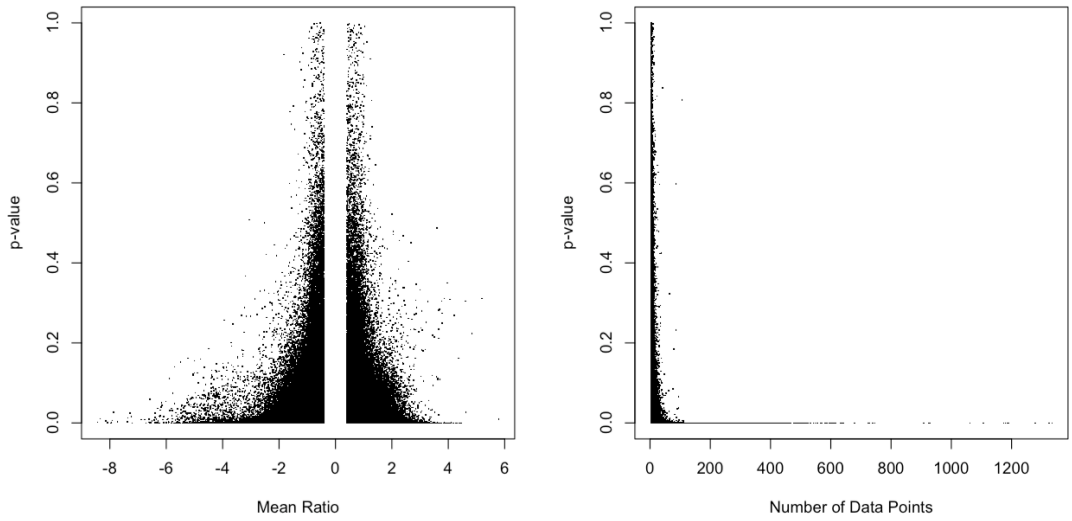


Figure 2-4 P-value generated by CNsolidate; Left - P-value vs. mean ratio, Right - P-value vs. number of data points.

Above (see **Figure 2-4**) the relationship between the assigned p-value, the mean ratio values (left) and the number of data points (right) of each segment across a number of different data sets is shown. Overall, high p-values are rarely observed where either the absolute mean ratio or the number of data points in a segment is high. This is expected as both the mean ratio and number of data points within a segment can act as proxies for segment quality in general.

CNsolidate aims to achieve high sensitivity while allowing the accurate ranking of detected segments. The local significance (p-value) is provided only as an additional detection filtering option. We suggest that the weighted confidence score (wscore) should be used as the primary filtering value.

Score Calibration

Using the detection scores defined above it is possible to rank detections within individual data sets. However, in some cases, it may still not be possible to set a single cut-off to yield a certain level of 'truth' across large numbers of data sets displaying variable qualities. To address this we have included a semi-automated score calibration method which allows the score to be calibrated across data sets based on a desired level of truth. To use this adjustment strategy one needs to first define both a measure of truth and an 'estimator'. These can then be used in concert to approximate an adjustment function, allowing a single cut-off to be defined on the 'filter' values to achieve a certain level of truth across data sets.

Score Adjustment

For the adjustment to approximate a baseline truth level across different estimator backgrounds we fit curves describing the behaviour of the truth vector compared to the filter vector for all estimator levels.

First, we fit a polynomial regression using the general model:

$$y^{(t)} = b_0 + b_1 x + b_2 x^2 + b_3 x^3 +, \dots, b_n x^n \qquad [2\text{-}33]$$

where $y^{(t)}$ is the filter value at truth level *t*, *x* is the estimator level and *n* equal the maximum order of the polynomial.

Next, we define the polynomial function f_t as:

$$\hat{y}^{\{t\}} = a_n x^n + a_{n-1} x^{n-1} +, \dots, + a_2 x^2 + a_1 x + a_0 \quad [2\text{-}34]$$

where *x* equal the estimator level, $a_0, a_1, a_2, \dots, a_n$ are the constant coefficients obtained from $\hat{y}^{\{t\}}$ and *n* is the maximum order of the polynomial.

Finally, we iterate over each discrete truth value *t* across the truth measure range *T* to define a vector of functions *V*:

$$V_x = f_t : \forall x \in T \quad [2\text{-}35]$$

In other words, we derive a set of functions describing the filter value, across the given estimator background, across the given truth measure range. This general approach can be applied to any combination of truth, filter and estimator vectors to approximate the filter value needed to obtain a certain level of truth at each discrete estimator value.

2.2.2 Copy Number Tagging SNPs

Assay Selection

A relatively small number of previously published studies [50, 104] on CNV have reported the observation that particular copy number variable regions (CNVRs) can be 'tagged' by a particular, normally nearby, SNP.

During the design of the DDD array-CGH arrays we have included 30 three component 'SNP-tagged' CNVRs from a study into population CNVs [50], additionally we have added the 30 SNP locations to the Sanger sample tracking system (sample logistics) via a custom Sequenom plex. This allows the required information to be generated such that the array-CGH data can be correlated with the assay run when the sample enters the building (sample logistics).

Copy Number Assignment

A CNV tagging SNP can be defined as a SNP genotype (or genotypes) that occurs at the given SNP location in the presence of a particular copy number state (or states) at the given CNVR more often than can be expected by chance. Correlating the observed SNP genotypes and copy number states for a given SNP and CNVR pair needs to be treated as a probabilistic estimate rather than the binary classification used for the sequencing and genotyping data tracking. This is mainly due to the fact that the frequency of observing a combination of SNP genotypes and copy number states at each given CNVR may not be observed at the same rate in all training data sets.

The first critical step is to assign a copy number state to each SNP-tagged CNVR for any given array-CGH data set. Each such CNVR contains three probes replicated five times on each of the DDD array-CGH array designs. To estimate the point at which copy number states change (log2 ratio space) we used the mixture model based approach available within the CNVtools R package [103]. CNVtools uses t-distributions to define a number of components from an input data distribution.

Log2 Ratio Clustering

To define copy number state boundaries we applied the mixture model, based on t-distributions, from CNVtools to the mean log2 ratio for each of 845 DDD control data sets at each of the 30 SNP-tagged CNVRs.

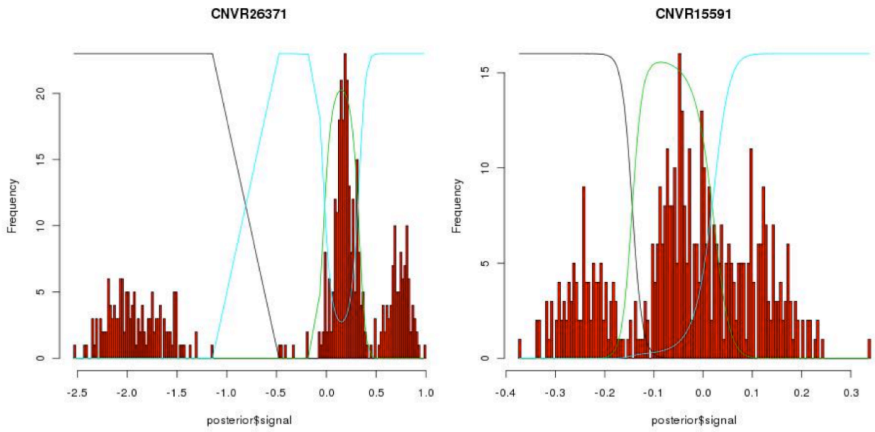


Figure 2-5 Copy number state genotyping. Left - Example of a 'well-clustered' CNVR. Right - Example of a 'poorly-clustered' CNVR.

Above (see **Figure 2-5**) are histograms showing two CNVRs included on the DDD array designs. For each CNVR the mean log2 ratio for each control sample and the estimated copy number state boundaries are plotted. CNVR26371 (left) displays a large dynamic range and well-defined copy number state boundaries whereas; CNVR15591 (right) displays a poor dynamic range and less accurate state classification.

Bayesian Approach to Data Tracking

To generate a probabilistic measure of 'sample similarity' for data tracking in the array-CGH analytical pipeline we calculate the probability of observing copy number states given the observed genotypes using a Bayesian approach.

For each CNVR:

$$P(s|g) = \frac{P(g|s)P(s)}{P(g)} \qquad [2\text{-}36]$$

where, s is the estimated copy number state and g is the genotype.

Then, for the overall measure:

$$P(S|G) = \frac{P(s|g)_1 + P(s|g)_2 +, \dots, +P(s|g)_n}{n} \qquad [2\text{-}37]$$

where, S is all estimated states, G is all observed genotypes and n equal the number of CNVRs included.

During the calculation of the overall similarity measure P(S|G) there is the opportunity to incorporate some additional information (weighted mean), for example, a measure of how well each CNVR could be clustered.

Note: the datasets used here often have relatively high numbers of missing values, to maintain a consistent scale we elected to use the mean in place of, for example, the product.

Empirical Parameter Estimation

To empirically derive the values of *P(g)*, *P(s)* and *P(g|s)* we used 1690 DDD control data sets. These data sets were generated for both DDD array-CGH array designs using 845 DNA samples from normal control individuals. Each individual was assigned a copy number state for each of the 30 included CNVRs and a SNP genotype from the appropriate tagging SNP. Then to define *P(g)* and *P(s)* we simply determined the frequency of observing each genotype and state respectively, for each CNVR individually.

To define *P(g|s)*, for each CNVR we determined the frequency of observing all possible genotype / state pairs at each CNVR individually.

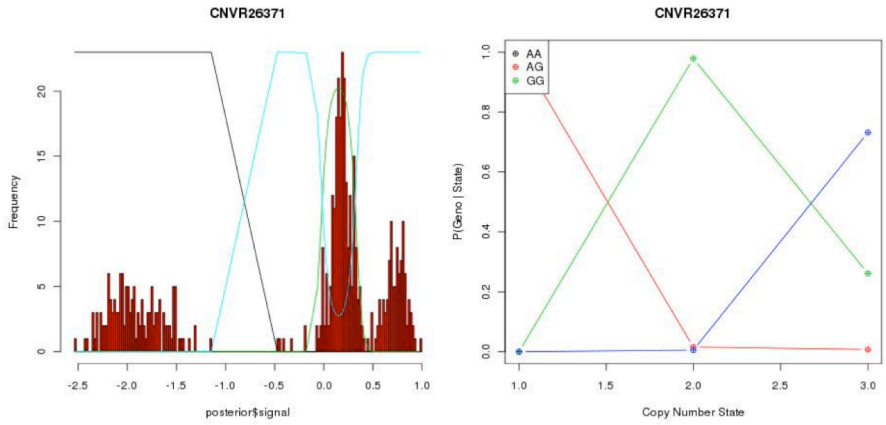


Figure 2-6 Copy number state frequency estimates. Left - Cluster plot of CNVR26371, Right - *P(g|s)* for each copy number state of CNVR26371.

Above (see **Figure 2-6**) are two plots showing the SNP-tagged copy number variable region CNVR26371. The left panel shows the derived copy number state boundaries and displays a good dynamic range. The right panel shows the estimated values of *P(g|s)* for each copy number state. We can observe across the 845 DDD control samples that:

- The copy number state 1 is observed in concert with the SNP genotype AG 100% of the time.
- The copy number state 2 is observed in concert with the SNP genotype GG the majority of the time, with the AG genotype having been observed in only a few cases.
- Copy number state 3 is more difficult to correlate with any SNP genotype, having shared its observations between the AA and GG genotypes.

Dynamically Updating Probabilities

This method uses a dynamic update for the estimates of *P(g)*, *P(s)* and *P(g|s)* as well as the definition of copy number state boundaries for each CNVR. The justification for this is that as the amount of data generated increases, the reliability of all these estimates should also increase.

This is quite a heavy analysis as each DDD data set comprises of approximately two million data points and the total number of samples for the project is expected to be greater than 10000. As a result every time an update is required, all data sets generated to date must be accessed and the relevant data points extracted to allow the required calculations. To allow this, and other analysis, to occur in a timely fashion we have developed a novel data accessing approach. These methods are available as a stand-alone R package called Rbin with no external dependencies. This package makes use of methods that can write C++ classes to file in binary format and allows super fast data extraction using random access. The novel part of this method is the fact that C++ data classes are written out in binary format and a single index and lookup is used to access any data range across multiple data sets sequentially or in parallel. The results of these accesses can be returned directly as large matrices in R, directed to standard output or written to a specified filename. Furthermore, any new data format can be simply defined and an index created via a number of intuitive function calls within the R interface. Additionally, these methods do not depend on any pre-existing package such as the Rcpp package and the Rbin interface between C++ and R is direct and fast (native).

To give an example of performance characteristics for the level of expected usage, running on a Intel (R) Xeon (R) 2.83GHz CPU with 2G RAM, to query a data range comprising of 1,903 data points across 10,000 data sets, the method will return a 1903 x 10003 matrix, containing 19,035,709 data points, to R within 9.75 seconds. To give an example of extreme usage, to query the entire chromosome 1 (172,380 data points) across 10,000 data sets the method will return a 172,380 x 10,003 matrix, containing 1,724,317,140 data points, to R within 17.42 minutes.

Both examples given here use the sequential file reads, the parallel data access methods can potentially deliver superior performance. However, due to the overall extreme speed of these methods, it is highly likely that the performance of file access on the file system itself will be the major speed-limiting factor.

2.2.3 CNV Consensus Reference Set

Genome Builds & Data Sources

All data sources denoted by * were obtained with GRCH36 (hg18) genomic mapping co-ordinates. These data sources were subsequently lifted over to GRCH37 (hg19) using the UCSC-Galaxy lift over tool [146]. Additionally, some data sources contained pre-merged CNV locations with frequency information, others contained pre-merged CNV locations with no frequency information and some contained raw call lists from individual samples. For my purposes it was most useful to obtain the raw call list format since it provided the greatest flexibility when defining merged CNV positions, frequencies and types across study sets.

Included Data Sets

- 42 Million study - raw call lists*.
- 42 Million study - genotyped regions*.
- WTCCC study - merged Affy6 data set*.
- 1000 Genomes pilot - merged deletions*.
- 1000 Genomes pilot - tandem duplications*.
- DDD study - national blood service controls.
- DDD study - generation Scotland controls.

DDD Control CNV Calls

The DDD arrays were designed against GRCH37 (hg19) genomic mapping positions. Detections where made across the two DDD control sets using CNsolidate with default settings and no hard cut-offs. Calls where then filtered based on two specific quality measures:

- Adjusted Wscore above 0.5.
- Local pvalue below 0.01.

All filtered calls from the DDD control sets were merged into a single list of CNVEs using an iterative reciprocal overlap rule (see **Below**). This set of common DDD control CNVs was then merged with the CNV consensus set v1 using the same approach.

CNV Merging - Study Sets

Copy number variable regions from the 42M study were obtained in the form of raw call lists (CNVRs) with no frequency information. These CNVRs were merged into an individual set of non-overlapping CNV events (CNVEs) containing frequency and 'state type' information using the same approach as the DDD CNVRs (see **Below**).

Raw Call Lists

First, we define the binary relationship *Re* as a 0.5 reciprocal overlap between two features within a set.

- Chromosomes are analysed separately
- Raw call lists are merged using *Re*
- The first two CNV features observed to share *Re* are grouped to form an initial cluster

- Furthermore any feature within the remaining chromosomal set displaying *Re* with any feature from within the cluster are also added.
- Finally, once no more feature displays the required *Re*, the break point positions for the cluster are defined to be 0.9 of the inner to outer break point distances within the cluster.

Additionally each raw call is tagged with a type such that each resulting CNV location can maintain its 'type state'.

For example, where all raw calls contributing information to a particular genomic position are only type1, the resulting merged genomic position is also type1. Where locations are a mix of types the associated type states are maintained. During this analysis the individual types are defined as -1, 1 or 0 to denote deletion, duplication and deletion / duplication status respectively.

Pre-merged Data Sets

42M Genotyped Regions

The 42M-genotyping study estimates a set of CNV genotypes (copy number states) across 450 samples for 4978 merged CNV regions (CNVRs). To generate the frequency information first we convert the CNV genotypes into a type state (either -1, 0 or 1) and simply count the number of samples that contributed to each type for each CNVR. Although the 42M genotyped genomic study is based on a subset of the 42M call list they are treated as a separate study due to the fact that it was run on a different array and used a different sample size.

WTCCC study Affy6

For the WTCCC study a detailed list of merged CNV locations with frequency information and the raw call lists for all 5,919 samples included in the study were obtained. The above method was used to integrate the merged deletion and duplication CNVEs to define type states of -1, 0 and 1. Then the merged break points of 0.9 between the inner and outer break point locations for each feature were used to define the CNV locations. Additionally, if two (or more) merged CNV features from the affy CNVE sets were merged, creating a feature of type state 0, the number of observations from each of the contributing CNVEs were summed to maintain the frequency information.

1000 Genomes Pilot

For the merged deletion set from the 1000 Genomes pilot project, a list of merged CNVRs with no frequency information was obtained. Therefore, while awaiting further information, we class each of these merged CNV locations as a singleton observation. The 1000 Genomes pilot tandem duplication data were not pre-merged so we applied the same rules as for feature merging of the 42M & DDD calls sets.

Assigning Common, Rare and Novel CNV Status

The overall goal of the CNV consensus set is to enable CNV detections to be accurately assigned to one of either common, rare or novel frequency status types. This is initially for direct use in filtering patient CNV detections from the DDD project but could also be applied to any study of CNV where frequency

information of commonly observed CNV locations are required. To this goal we provide two different approaches to CNV status assignment.

Approach 1 - Individual CNVE Sets

In this approach any 'test' CNV location is compared against all individual CNVE sets from the CNV consensus. This allows for different parameters to be chosen for individual CNVE sets.

For example, consider where two sets from the CNV consensus shows quite different CNV characteristics (e.g. affy vs. 1Gdup). Using this approach has the benefit that different criteria for both defining similarity (overlap and type status) and frequency binning (which frequency relates to common, rare and novel) can be set for different CNVE sets.

Approach 2 - Combined CNV Consensus Set

In this approach all of the component CNVEs sets are merged into an overall CNV consensus track. We choose to treat all CNVE sets identically during the merging steps and applied the standard 0.5 reciprocal overlap and 0.9 break adjustment parameters. We choose the maximum frequency between the contributing CNV locations from the individual CNVE sets as the frequency estimation for the resulting merged CNV consensus location. Type states of -1, 0 and 1 were maintained as previously mentioned.

The major benefits of this method compared to approach 1 are:

- The CNV consensus set is now a single data source containing CNV positional, frequency and type information.
- The type state information is determined across sets so the CNV type definition is more accurate from the consensus than the individual CNVE sets.
- The break point information of merged CNV locations across CNVE sets is likely to be more accurate due to the larger number of observations contributing to the consensus CNV location.
- It is easier to maintain and distribute the CNV consensus set in this format.
- Adding extra information into the consensus is more straight forward and does not require extra code / parameter definitions to deal with new CNVE sets.
- The display of data as tracks in data browsers is more intuitive and easier to implement.

The CNV consensus reference set can be made available in flat txt file format containing all the required information for use in filtering genomic variants. However, it is also contained inside the CNsolidate package along with a number of intuitive functions to allow filtering using a number of different approaches. For the DDD project there is a dedicated SQL analysis database which stores information about each analytical process as data runs through the pipelines. This database has a number of tables that relate to various flavours of variant filtering including the current CNV consensus reference set.

2.2.3 CNV Filtering

Overview

We have developed a CNV ranking and prioritisation system in the DDD project for flagging CNVs of potential clinical relevance [147]. A major element of the DDD project is to facilitate a genetic diagnosis for patients and families who have been recruited into the study. CNVs meeting certain criteria relating to clinical interest are flagged by an automated filtering system prior to being reviewed in detail during a weekly multidisciplinary reporting meeting.

DD Gene to Phenotype Database

The CNV filtering pipeline makes use of a valuable resource called the DDG2P (dd gene to phenotype) gene list that is downloadable from the DECIPHER database. The DDG2P is a manually curated database of gene to phenotype relationships containing primarily genes with some association to developmental disorders. The database contains gene names, genetic mechanisms, mutation consequences and linked phenotype terms. Each DDG2P entry is placed into one of four possible categories [“Confirmed DD Gene “, “Probable DD gene “, “Possible DD Gene “, “Both DD and IF”] based on the amount of evidence available for the described association. Each gene is associated with specific developmental phenotypes or syndromes via particular genetic mechanisms (autosomal dominant, autosomal recessive and X-linked) and mutation consequences of the gene product (loss of function, activating mutation, increased gene dosage, etc.) Using DDG2P enables any rare variant in known DD genes with a predictable effect on the gene product to be flagged on the basis of inheritance, genotype and likely mutational consequence.

Not all variants in genes known to be in association with one or more genetic disorder result in the phenotypic display of the associated symptoms [148]. One reason for this is that the specific genetic mechanism needs to be considered. Other reasons for can include more complex situations such as dosage compensation [149], partial penetrance [150] and digenic and polygenic effects [151]. The linked phenotypes included in the DDG2P database use the human phenotype ontology (HPO) [152-155] to describe phenotypic traits. The use of a standard ontology and vocabulary for describing patient symptoms is key to the accurate phenotyping of patients in DDD study. As patients are recruited into the DDD study across the 24 different regional genetic services and each centre has a number of practicing clinicians, differences in phenotyping detail and quality is unavoidable. By using a fixed ontology these differences are minimized by forcing everyone to follow the same fixed set of terms and avoid problems relating to free text parsing. Each entry in the DDG2P has a number of linked HPO terms believed to be in association with the variant in the gene, however it was decided not to include any phenotype matching in the filtering pipeline. Instead all variants matching the mode and mechanism in any of the confirmed and probable DDG2P entries are flagged. Such variants are normally of general clinical interest and therefore it is important to prioritise them for clinical review as subtle differences in patient phenotypic display may not always be recognized immediately by a referring clinician.

Filtering Rules

I have implemented a rule-based approach to CNV filtering and prioritisation for clinical review. This filtering package includes three main steps:

1 Sample and Call QC criteria
2 Variants in Developmental disorder genes
3 Variants of uncertain significance (VOUS)

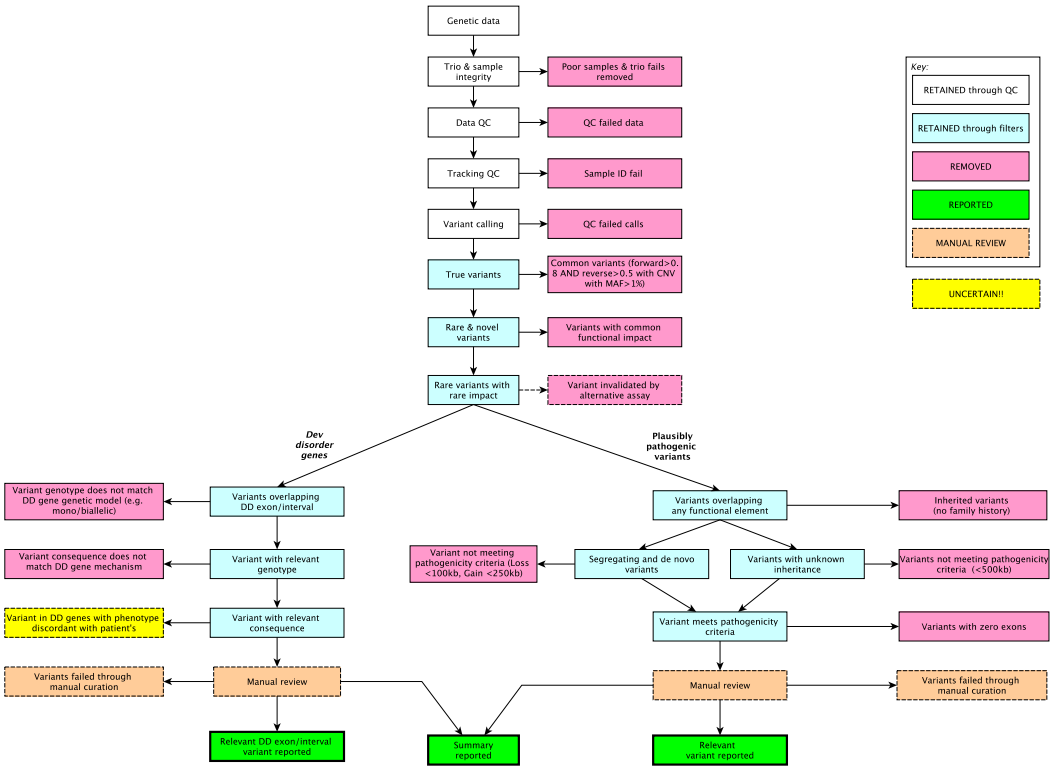


Figure 2-7 Flow diagram of the CNV filtering pipeline rules.

Above (see **Figure 2-7**) shows an overview summary of the CNV filtering pipeline for flagging CNVs of potential clinical relevance. White boxes denote QC steps, Blue boxes denote CNV call filtering steps, pink boxes show steps where samples and calls are removed, green boxes denote areas where reporting of variants become possible and orange denotes area where a manual review is required.

Sample and Call Quality Control

Every dataset run through the CNV clinical filtering pipeline needs to pass a number of quality control (QC) parameters. Most importantly the dataset needs to pass the CNV sample-dataset tracking method previously described. This is the most important QC step and ensures that the dataset under consideration is linked to the correct patient sample. On top of the data tracking QC a number of data quality QC checks are applied to ensure adequate data reliability. For all datasets the data quality QC is applied as an exclusively post CNV detection

method. Perhaps the most informative predictions of CNV call quality for an entire sample can be assessed only once CNV detection has actually taken place.

For sample QC we use an exclusively post-calling approach and apply a robust clustering algorithm ("aberrant") for outlier identification and exclusion [156] to the total number of passed CNVRs per sample and the proportion of passed novel CNVRs per sample. We define novel as CNVRs that do not share greater than 80% of their boundaries with a copy number event (CNVE) of the same type, deletion or duplication, contained within the CNV Consensus list. Each array-CGH sample is made up of two slides (2x 1 Million probe Agilent arrays) and any slides defined as outliers by the aberrant clustering method are failed and where either slide fails for a sample the overall sample is also failed (see **Figure 2-8**).

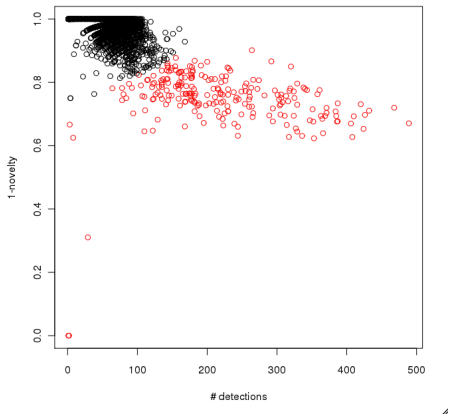


Figure 2-8 The number of CNV calls per array-CGH slide vs. the proportion of novel CNV calls per slide.

Above (see **Figure 2-8**) the results of applying the abberant method to the number of CNV detections against the proportion of novel CNV detections per slide is shown. Data points shown in red are assigned outlier status by abberant and are failed. For each sample, if either array-CGH slide fails data quality QC the entire sample is failed and a repeat array-CGH experiment is requested.

Additionally further exclusions, at the sample level, are applied, based on a low sensitivity cut-off of less than 40 QC passed detections and a deletion / duplication ratio of greater than 10. Finally we use the data tracking check to ensure that the CNV data is consistent with the Sequenom data linked to the same sample identifier. Using the 28 copy number tagging SNPs allows the array-CGH data to be correlated with the SNP genotypes obtained via the Sequenom assay run at sample reception. Both slides on the array-CGH platform contain dedicated probes tiling the 28 CNVRs tagged by SNPs present on the Sequenom assay. First, CNVtools [103] is used to assign a copy number state to each SNP-tagged CNVR and the probability of observing all copy number states (array-CGH data) given all observed genotypes (Sequenom data) is calculated using the Bayesian framework previously described.

For call QC of the array-CGH data we apply the recommended detection quality filtering criteria from CNsolidate, comprising of a wscore above 0.5, a p-value below 0.01, and the mean log2 ratio below -0.41 for deletions and above 0.36 for duplications when passing individual CNV calls.

A number of additional call QC parameters for CNVs applied to the clinical filtering pipeline are applied. All CNVs applied to the clinical filtering pipeline must be rare and exonic; we define rare as CNVRs that do not share greater than 80% of their boundaries with a copy number event (CNVE) of the same type, deletion or duplication, observed at more than 1% in the CNV Consensus list. To determine whether a CNV is exonic, i.e. contains at least one exon within its boundaries, we use the GENCODE [139] gene set (version 17). CNVs are defined as exonic if at least one exon from version 17 of the GENCODE gene list overlaps the CNVs boundaries by at least 1 base pair (any overlap).

Finally a stringent quality measure to CNV calls that we term the MADR (mad region) is applied, the MADR is a measure of the relative difference in response (mean log2 ratio) and regional noise (MAD) across samples. To calculate the MADR value estimate the noise of the CNV region across samples is needed. To do this we make use of the "Rbin" fast data access package previously described. For each detected CNV the log2 ratio values for each probe within the CNV across all array-CGH datasets available are extracted. Then the mean log2 ratio (*meanl2r*) for each sample is used to calculate the median absolute deviation (*MAD*) of the mean log2 ratio value across samples. This results in a single cross sample noise measure (*mad_region*) for each CNV. The MADR value is then calculated thus:

$$MADR = abs(\frac{meanl2r}{mad\_region}) \quad [2\text{-}38]$$

The MADR value acts as a stringent cut-off on CNV call quality and also as a proxy for common variation. It is a measure of the relative difference in probe response (mean log2 ratio) and regional noise (MAD) across samples. CNV calls with high MADR values are far away from the background noise of the region and are highly likely to be genuine CNV events. Furthermore CNV regions with high noise across samples are likely to contain relatively large numbers of common CNV events. These regions may not always be included in the CNV consensus reference set in high enough numbers to be classed as a common CNV region so this additional stringent QC measure may also filter out poorly understood common CNV regions.

As implied this additional QC measure has high stringency and may filter out a number of interesting, clinically relevant CNV events, however the decision, in the first instance, was to be very cautious for flagging CNVs of potential clinical interest. However the clinical filtering pipeline has been designed to be iterative in nature and can be easily rerun with different parameter definitions.

Flagging CNVs for Clinical Review

To assign a filter status for CNVs and flag potentially clinically relevant CNVs for clinical review we need some specific information about the sample, CNV and gene. Due to the fact that no phenotype matching (HPO terms) is being

attempted the only sample level information needed is the stated gender of the sample. To obtain this information the sample ID (sanger_id) is used to query the DDD LIMS system to obtain the decipher_id (patient identifier for the decipher database), the stated gender is then compared to the genotyped data from the array-CGH sample and if they match the filter will continue otherwise an automated email is sent to the DDD informatics team. The CNV level information is all contained within each CNV call and the gene level information is contained inside the DDG2P database.

Developmental Disorder Genes

For flagging CNVs in DD genes we compare the mode and mechanism of confirmed and probable DD genes to the chromosome, copy number state and gender of rare, exonic QC passed CNV detections.

Table 2-1 Shows the rule-based approach to CNV filtering for clinical relevance based on the DDG2P gene list.

CHR	CNS	SEX	MODE	MECH
1,24	0	M,F	Biallelic	Uncertain, Loss of Function, Dominant Negative
1,24	0,1	M,F	Monoallelic	Uncertain, Loss of Function, Dominant Negative
1,24	>2	M,F	Monoallelic	Uncertain, Increased Gene Dosage
23	0,1	M,F	X-linked Dominant	Uncertain, Loss of Function, Dominant Negative
23	>2	M,F	X-linked Dominant	Uncertain, Increased Gene Dosage
23	0,1	M	Hemizygous	Uncertain, Loss of Function, Dominant Negative
23	>2	M,F	Hemizygous	Uncertain, Increased Gene Dosage

The rules needed for flagging CNVs based on the DDG2P and CNV annotations can be collapsed into certain combinations of chromosomes, copy number states, modes and mechanisms (see **Table 2-1**). DD genes with a biallelic mode on any chromosomes, in both males and females with either uncertain, loss of function or dominant negative mechanisms are always flagged if, and only if, the estimated copy number state is zero (i.e. there is no predicted functional copy of the gene). For monoallelic DD genes firstly the rules are exactly the same except that single copy loss (heteroygous deletions) are also flagged. Secondly, monoallelic DD genes CNVs with a copy number state greater than 2 (gains) are flagged in they are found in DD genes with either uncertain or increased gene dosage mechanisms. DD genes with an X-linked mode of action follow exactly the same rules as for monoallelic DD genes except clearly the CNV must be present on chromosome X. Finally for DD gene where the mode of action is defined as hemizygous, CNVs resulting in a predicted decrease in gene dosage, those with either uncertain, loss of function or dominant negative mechanisms, are flagged if the sample is male and if the copy number state is either zero or one (i.e. the predicted dosage of the gene has been decreased). Note that the reference sample is a pooled reference make up of 500 male individuals. For predicted gene dosage increases in hemizygous DD genes CNVs are flagged in both male

and female samples if the copy number state is greater than 2 and the DD gene mechanism is either uncertain or increased gene dosage.

Variants of Uncertain Significance (VOUS)

As well as flagging CNVs in genes whose function is well understood and which have been linked with developmental delay (DDG2P), CNVs whos contribution to clinical phenotypes is unclear are also flagged. The class of CNVs flagged using these rules are sufficiently rare in the general population such that they are normally of interest to the referring clinician even if their contribution to clinical phenotypes is poorly understood. To flag these types of variants a number of fixed cut-offs on rarity and size given the inheritance classification for each detected CNV are used. As mentioned previously a cut-off of less than 1% population frequency is used to define CNVs as rare and additionally the MADR values are used as a CNV call quality measure and proxy for common CNVs.

Table 2-2 Size cut-offs for different CNV inheritance classifications and parent-affected states.

Inheritance	**Parents Affected**	**CNV Size (Kb)**
paternal	Father	DEL=100, DUP=250
maternal	Mother	DEL=100, DUP=250
de novo	Any	DEL=100, DUP=250
biparental	Any	DEL=100, DUP=250
unknown	Any	DEL=500, DUP=500

For all rare CNVs detected we apply three size cut-offs based on the inheritance classification of the CNV (see **Table 2-2**). For all CNVs where the inheritance classification is unknown we use a fixed cut-off of 500kb and any rare CNVs (losses or gains) greater than 500kb in length are flagged for clinical review. For CNVs where no informative inheritance classification was made, if the inheritance is '*de novo*' or biparental the fixed size cut-off used for deletions is 100kb whereas for duplications it is 250kb and any rare *de novo* deletion or duplication greater than 100kb or 250kb in length respectively are flagged. For patients who have either, or both parents affected, paternal and maternal inherited CNVs can be flagged so long as the father or the mother is affected, respectively. If the inheritance is classified as biparental and both parents are affected. Additionally male patients CNVs on chromosome X classified as maternal are flagged irrespective of the mother's affected status.

2.3 RESULTS

2.3.1 CNsolidate

Naive Voting

One obvious way to increase one's belief that individual change point detections may be robust is to naively consider how many algorithms are in agreement at the given change point location. This is a major benefit when using multiple algorithms and can be classed as a type of voting system (naive voting in this case). The figure below (see **Figure 2-9**) shows, for eight algorithms, the relationship between the number of algorithms that are in agreement and an estimate of the Type I and Type II error rates for all detections made across a single data set.

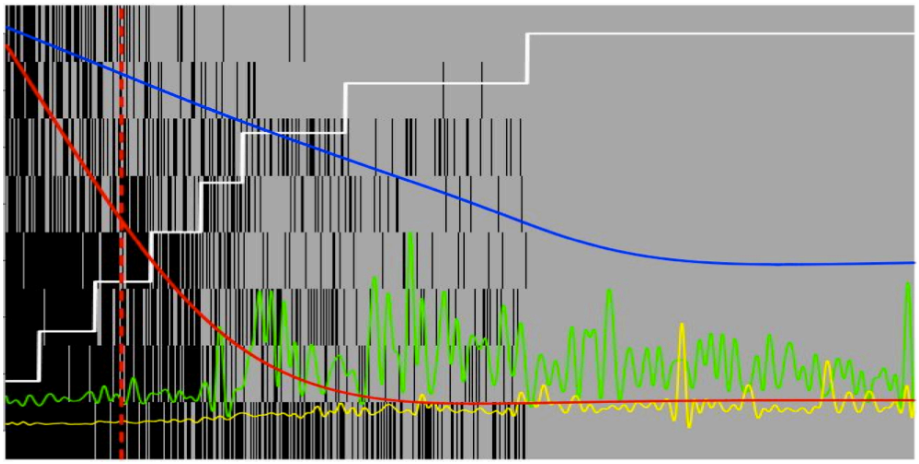


Figure 2-9 Naive Voting Example - CNsolidate version 1. The background color denotes where detections were made (grey) and where they were not (black). The white line shows the number of algorithms in agreement, ranging from 1-8, left to right. The red line is an estimate of the Type I error rate using the CNV consensus set as the gold standard. The blue line is an estimate of the sensitivity assuming that all detections equal a sensitivity of 1. The yellow line represents the size of the detection and the green line the mean log2 ratio (response). The dashed red line shows the point on the graph were the FPR is closest to 0.05.

The algorithms are ranked, top to bottom, based on the total agreement across all detections (see **Figure 2-9**). This is an example of CNsolidate version 1 and in this case the recommended number of voters is 3. CNsolidate version 1 uses a naive voting system, where all voters are considered to add equal weight to CNV detection call quality. Although this can be effective it is not the optimum approach to a voting system. Consider the case where one particular voter becomes unreliable given a particular class of data. Better to approximate the behaviour of each and every voter given certain measurable characteristics of the underlying data ('expert voting').

Expert Voting

The current version of CNsolidate uses an expert voting system in place of the naive approach. Rather than simply counting the number of algorithms in agreement at each change point detection, we derive a measure of detection quality (wscore). This value (wscore) is derived using the estimated performance of each algorithm given certain predictive variables that can be drawn from the input data characteristics (see **Methods**).

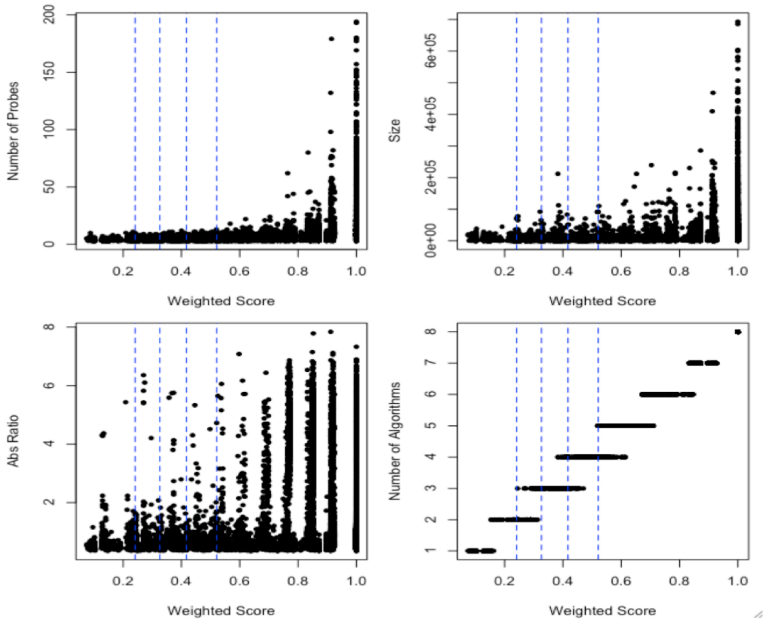


Figure 2-10 Weighted Score vs. certain predictive variables. Top left - wscore vs. number of probes. Top right - wscore vs. size (bp). Bottom left - wscore vs. absolute mean ratio. Bottom right - wscore vs. number of algorithms.

As indicated the wscore displays a range between 0 and 1 with increased detection quality tending towards 1 (see **Figure 2-10**). Although the number of algorithms in agreement is taken into account, due to the nature of the wscore calculation (weighted mean) the overall value is based on the estimated performance of each contributing algorithm given a number of predictive variables. Above (see **Figure 3.2**) the raw (non-adjusted) wscore plotted against a number of predictive variables is shown. Generally, as the wscore value increases so does the value of all predictive variables. This is expected as the predictive variables are thought to act as proxies for detection quality overall. However, there is some subtlety within the wscore values relating to individual algorithm performances given all the predictive values used during the calculation. We propose that this subtlety yields finer grain detection scoring than any of the individual predictive components individually.

Score Adjustment

Having defined a detection quality measure (wscore) for CNV detection in individual data sets it is possible to calibrate the score across multiple data sets

given some measure of 'truth'. To that goal we have developed an adjustment strategy (see **Methods**). Using this method it is possible to adjust the wscore values given any estimator value for any desired level of truth.

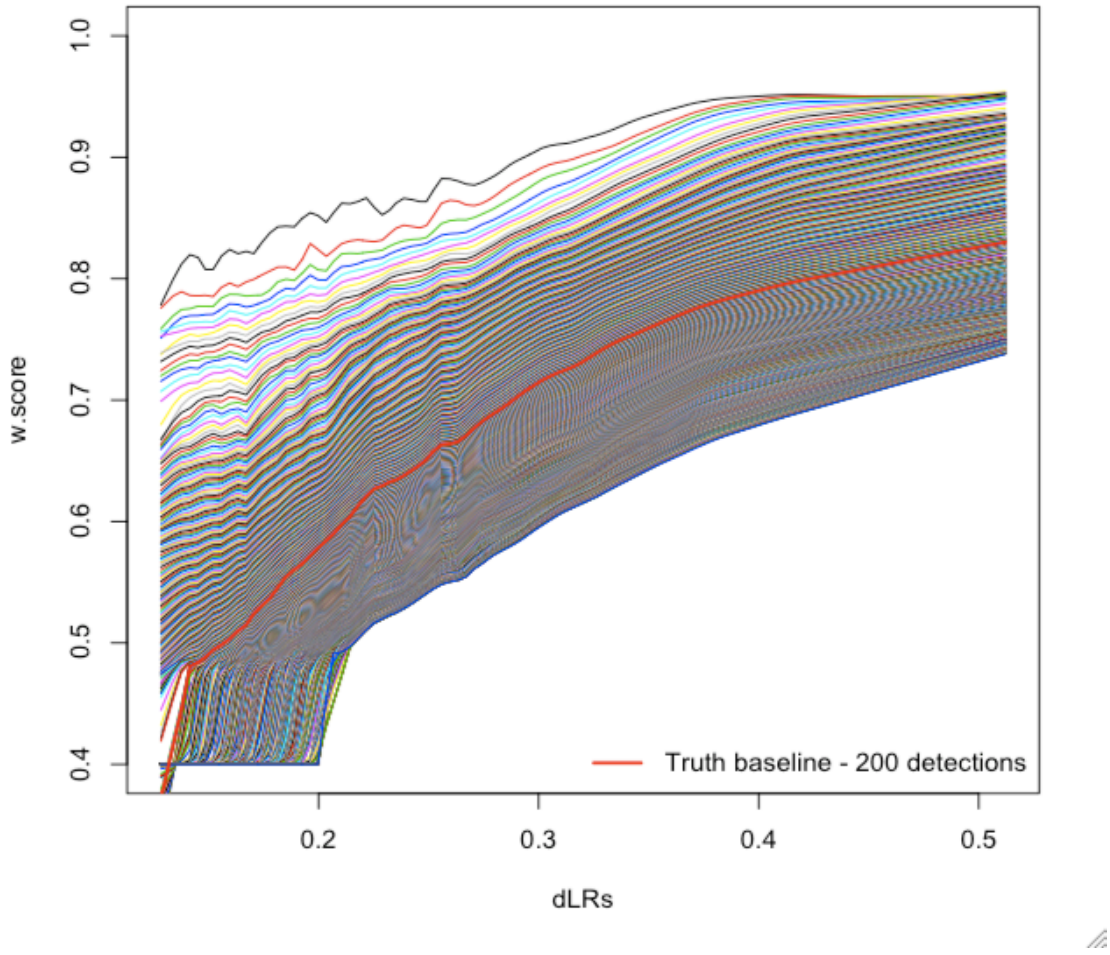


Figure 2-11 Curves representing adjustment functions at discrete levels of truth using the dLRs noise value as the estimator. Predictor value (dLRs) vs. estimated wscore adjustment value.

Above (see **Figure 2-11**) are some example curves, representing the derived adjustment functions for different levels of truth. In this example the total number of detections was used as the truth measure and a noise value (dLRs) as the estimator.

The truth level is set at increments of 1, between 10 and 1200 detections. Thus, for each of the functions, the truth baseline is relaxed across the noise range. The solid red line represents a truth baseline 200 detections.

The default adjustment function within CNsolidate is based on a novelty target of 0.25 using the dLRs noise measure as the estimator. The different levels of

'novelty' are defined via comparison against a gold standard reference set (CNV consensus set excluding the DDD controls) and the 0.25 novelty target is an adjustable parameter (see **Figure 2-12**).

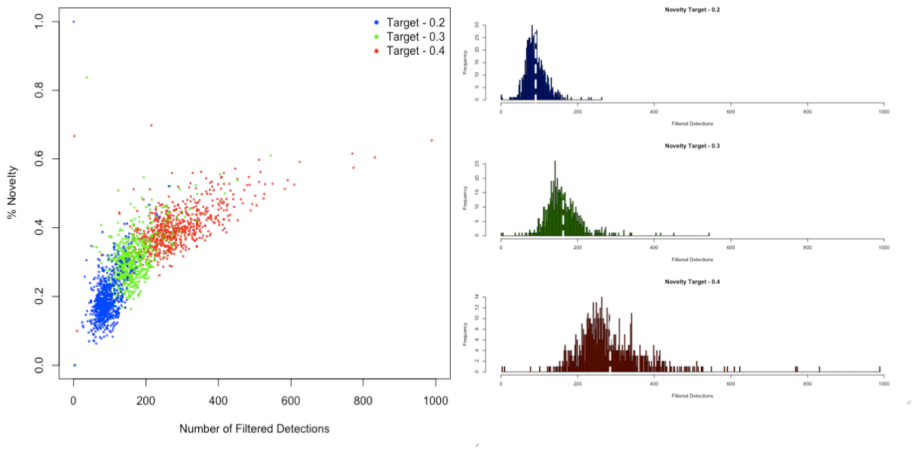


Figure 2-12 Applying different novelty targets to the DDD control data sets. Left - Number of detections passing adjustment value vs. percentage novel. Right - Histograms of percentage novel at three different adjustment function levels. Red - 0.4 novelty target. Green - 0.3 novelty target. Blue - 0.2 novelty target.

Above (see **Figure 2-12**) are plots showing the difference between applying a 0.2, 0.3 and 0.4 novelty target to the 845 DDD control data sets. The novelty adjustment functions were defined using the DDD controls as the training set. Therefore it is not too surprising that the adjustment functions perform rather well, adjusting the majority of data sets close to the desired level of novelty.

There is a certain degree of variance around each novelty target across the control data sets. This is expected as the adjustment function was defined using the data sets as a whole, not individual data sets. As the novelty target is relaxed, from 0.2 - 0.4, the variance in the number of detections across the data sets increase. This is not unexpected as by relaxing the truth target the adjustment functions are certain to become less reliable. This is due to the fact that, in the majority of cases, as truth targeting becomes more relaxed the number of options (wscore cut-offs in this case) for individual data sets to reach the desired level of truth is likely to be increased.

Technical Replicates

Sample Selection

A useful way to assess the consistency of any change detection method is to use technical replicates and look at concordance rates. In theory when the same DNA sample is applied to an assay, microarray in this case, the result of any change detection should be repeated across replicate experiments. Although there may be slight differences due to experimental factors or data artifacts the overall results should be consistent. The concordant results across replicate

experiments of the same sample can be thought of as true detections or true positives (TP). To put it another way the discordant results can act as a proxy for the false positive (FP) rate.

The DDD high thoughput array-CGH laboratory makes use of a technical replicate in the form of the well known HapMap sample NA12878. This DNA sample has been used as a technical control in a number of previous studies [157-162] including microarray and sequencing based projects. In the DDD the technical replicate NA12878 is included on every 96 well plate and acts as a check for data quality. This sample is known to perform well on microarray assays and if the sample fails data QC this can indicate a problem with the entire plate. During this section we make use of 50 replicates of NA12878 from both DDD array designs to allow us to estimate the false positive rate (or Type I error) for CNsolidate and its default 8 component algorithms ('voters'). These replicate data sets were not used during adjustment function training and so should be considered as a true test set.

False Positive Rate Estimation

To enable an estimate of the false positive rate using technical replicates and allow us to plot receiver operator curves (ROC) two parameters need to be defined.

A measure of truth and of segment similarity:

- A segment must be observed in at least 80% of the technical replicates (40/50) to be classed as a true detection.
- Two segments must share a reciprocal overlap of 0.5 or greater to be classed as the same detection.

This is a reasonably strict definition of truth and should ensures that the detections classed as true are highly reliable within the data set under question (50 replicates of NA12878). All results described during this section use this definition of truth.

Component Algorithms

The default option for CNsolidate is to use its eight highest performing algorithms for detection. The remaining four algorithms are considered to be less reliable and add little or nothing to the overall performance of CNsolidate. Additionally, running multiple algorithms has a cost in terms of speed and computational resources. This default option can easily be adjusted within the configuration settings for the CNsolidate package.

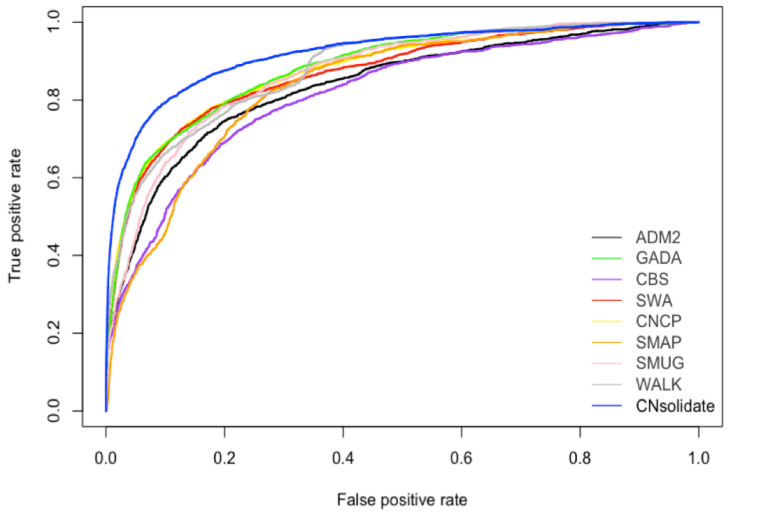


Figure 2-13 ROC of CNsolidate and its default algorithms. False positive rate vs. true positive rate for the eight component algorithms.

For any analysis of change detection the ultimate goal is to minimise the false positive rate (Type I error) while maximising the true positive rate (1- Type II error). When looking at receiver operator (RO) characteristics, in general, an increased area under the curves denotes situations in which performances are improved.

Above (see **Figure 2-13**) are RO curves for CNsolidate and each of its eight highest performing algorithms. CNsolidate displays superior RO characteristics compared to any of its component algorithms. Additionally, there are some differences in the RO performance of individual algorithms, however none display overall poor RO characteristics indicating that each algorithm has been reasonably well tuned.

RO Characteristic of the wscore

CNsolidate aims to improve detection rates, allowing a larger number of detections to be made, while providing accurate ranking of detection qualities within and between data sets. To that goal we have developed the weighted confidence score (wscore). The overall aim of the wscore is to allow a single cut-off to be set across multiple data sets to achieve a consistent level of truth.

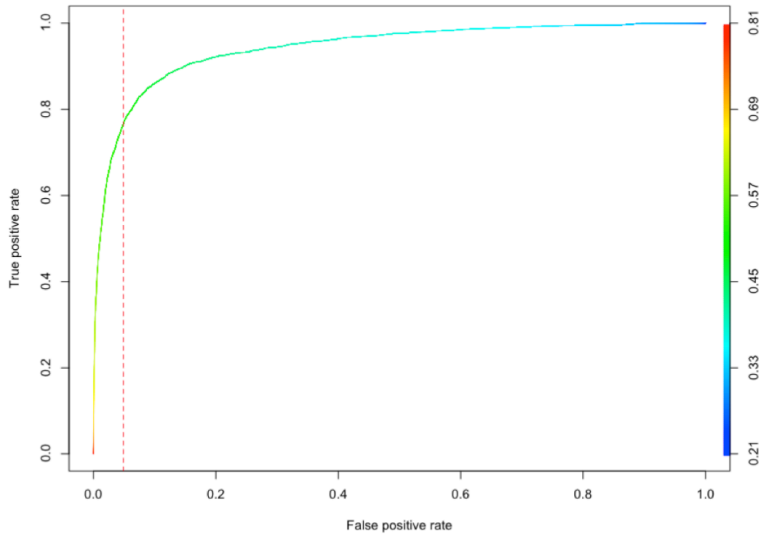


Figure 2-14 RO characteristic of CNsolidates wscore. False positive rate vs. True positive rate across the wscore space.

Above (see **Figure 2-14**) shows the same ROC curve as figure 2-13 however the colour of the line relates to the wscore and the dashed red line denotes the defined cut-off (0.5 wscore). The adjustment function, by definition, results in the wscore cut-off being set at 0.5. The scale of the wscore across the RO space is highly optimal, with increased values decreasing the false positive rate. As the cut-off on the wscore is relaxed the true positive rate increases sharply while the false positive rate increases gradually.

Using the defined cut-off (0.5 wscore) CNsolidate achieves a true positive rate of approximately 0.8 with a false positive rate of approximately 0.05. During studies of CNV using microarray based assays a false positive rate of 0.05 is generally considered to be acceptable. Furthermore, in our hands, the overall concordance rate of any two replicate microarrays is approximately 0.8, suggesting that the true positive rate reported here is close to optimal.

RO Characteristic of Other Predictive Variables

An approach often taken to change point detection from microarray-based data is to run an algorithm to detect variable data segments and then filter the detections based on various factors (e.g. mean ratio & number of probes). CNsolidate incorporates these, and other, variables into the calculation of the wscore. Below (see **Figure 2-15**) are further RO curves displaying the effect of using some different variables as the predictor for segment quality in terms of RO space.

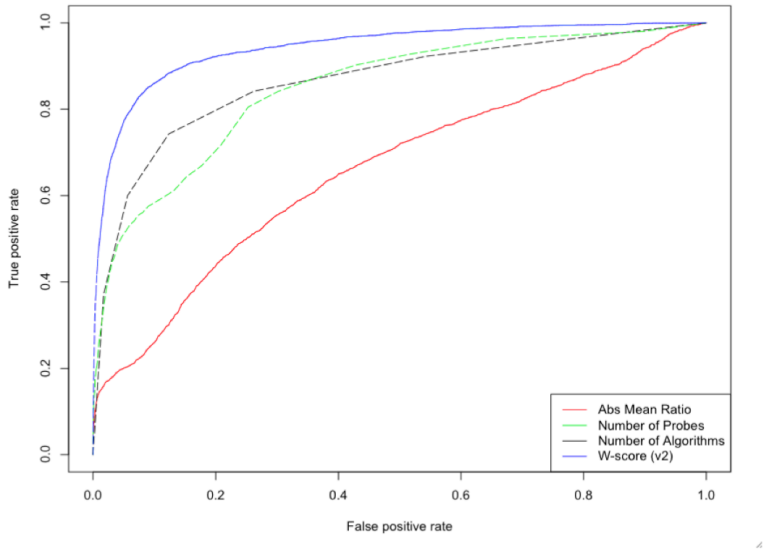


Figure 2-15 RO characteristic of wscore vs. some other variables.

Clearly when looking at the RO characteristic of the variables, the wscore is best for predicting detection quality and displays the best performance across the RO space (see **Figure 2-15**). Interestingly the mean ratio displays rather poor RO characteristics with the false positive rate increasing rapidly above the 0.2 true positive rate. The number of data points (probes) seems to be a far better predictor of detection quality than the mean ratio.

The naive voting system (number of algorithms) displays the second best RO characteristic. The difference between the number of algorithms and the wscore predictors denotes the improvement in detection quality scoring when using the expert compared to the naive voting systems within CNsolidate. This difference is a highly significant improvement. Furthermore, any improvements tending towards the upper end of the RO space (high performance) quickly become much harder to achieve.

2.3.2 Copy Number Tagging SNPs

The method described earlier (see **Methods**) for the tracking of array-CGH data is incorporated as a final check in the high-throughput array-CGH analytical pipeline for the DDD project. This is the last chance available to detect any tracking problems during the sample or data flow throughout the experimental and analytical pipelines.

Sample Swap Detection

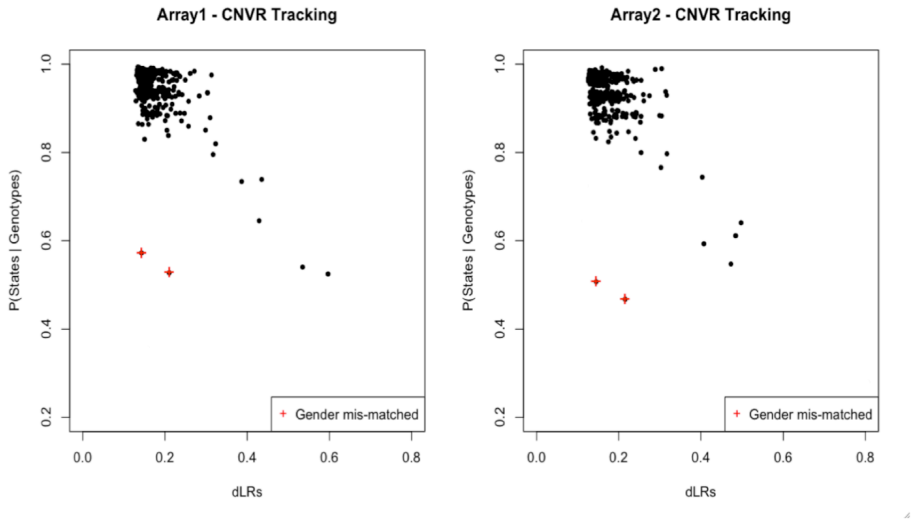


Figure 2-16 Data Tracking Values vs. Noise in the DDD Control Data Sets. Left - DDD array design1. Right - DDD array design2. Red - Potentially mismatched samples.

Above (see **Figure 2-16**) are plots for the data tracking values, P(S|G), against a measure of data noise (dLRs) across all DDD control samples for both DDD array designs. The majority of data sets display a posterior probability of a sample match of greater than 0.8. There is a clear trend displayed that when data noise levels increase (larger values of the dLRs measurement) the posterior data tracking values tend to decrease. This is understandable as the tracking values rely on an accurate estimate of copy number state, derived for the mean log2 ratio, at a number of different CNVR locations. As the variance of all log2 ratio values increase (increased noise) the estimate of copy number state becomes less reliable. The two data sets highlighted with a red cross are samples that were previously flagged by the laboratory as a potential sample swap.

Sample Swap Resolution

The two data sets (samples) highlighted previously both display only moderate noise measurements (dLRs < 0.2) however the data tracking values in both cases are low (posterior probability < 0.6). Additionally there was a mismatch between the expected gender and that reported from the array (derived from the mean log2 ratio on the X chromosome) in both cases.

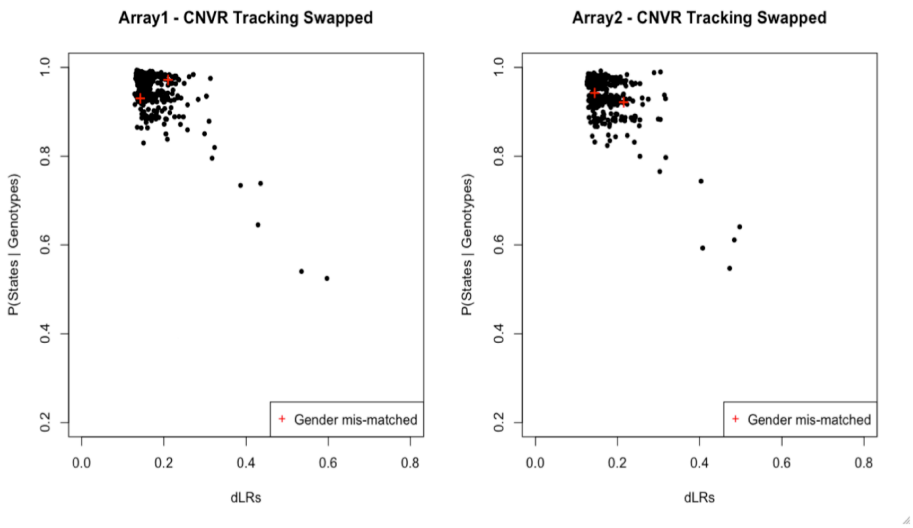


Figure 2-17 Swapping the aCGH data for two potentially mismatched samples. Left - DDD array design1. Right - DDD array design2. Red - Potential mismatched samples.

Normally, samples failing the tracking values but passing all other QC (e.g. data noise levels) would be failed immediately and a new sample would be requested. However, due to both the fact that these are control samples (not to be reported to the clinical teams) and that the samples were flagged by the laboratory as a potential sample swap it was possible to resolve the swap using the data tracking method. To do this we simply swapped the sample IDs for the array-CGH data and re-run the data tracking approach (see **Figure 2-17**). This process extracts the SNP genotypes typed on the Sequenom assay in sample logistics from an oracle database and then compares the SNP genotype against the estimated copy number state at each of the CNVRs using the method described previously (see **Methods**). As a result of switching the sample IDs on the array-CGH data sets above, the data tracking values were increased above the defined cut-off and the reported genders were no longer mismatched. This example acts as a nice positive control and demonstrates the utility of the data tracking method of resolving sample swaps based on CNV tagging-SNPs.

Sample Swap Discrimination

Having developed the previously mentioned method for array-CGH data tracking it is necessary to define a cut-off on the tracking values to discriminate between a sample match and a sample mismatch. To do this the DDD control data sets were used to look at the difference and discrimination power of the data tracking values by generating a real and a null distribution of tracking values.

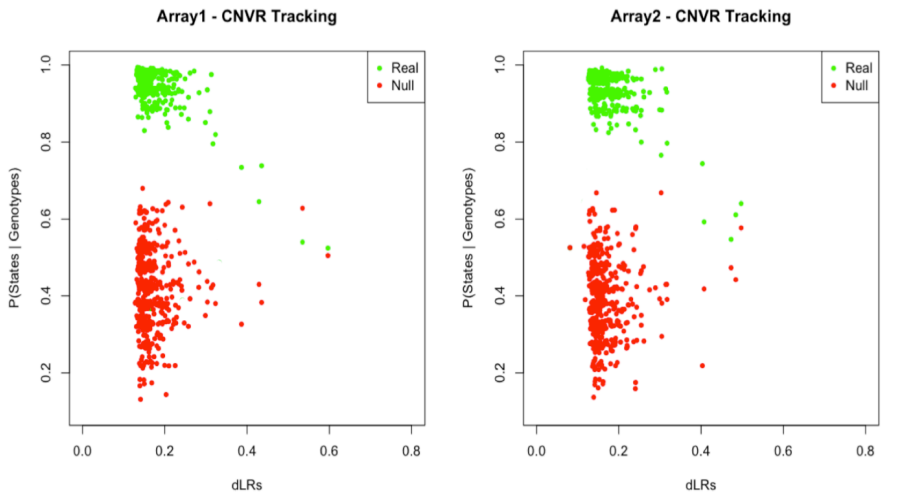


Figure 2-18 Expected vs. Null distributions of the data tracking values. Left - DDD array design1. Right - DDD array design2.

Above (see **Figure 2-18**) the DDD control data tracking values (having resolved the sample swap mentioned previously) for both DDD array designs are shown. The points plotted in green are the data tracking values using the sample ID supplied from the laboratory management system (the real distribution). The points plotted in red are the data tracking values after offsetting the sample IDs by one, resulting in all sample IDs being 'incorrect' (the null distribution). The two distributions (real & null) are quite different, displaying considerably different means and variances. The difference between these distributions denotes the discriminatory power of the data tracking values. Using this observation we have initially set the data tracking failure cut-off at a value of 0.7. However, as more data are processed, the estimate of both the probabilities and the copy number state boundaries should improve using our dynamic probability updating approach (see **Methods**).

Due to the fact that the data generated from an array-CGH experiment are very different to that produced from the Sequenom assay our method is considerably more complicated than that used for the genotyping or sequencing pipelines at the Wellcome Trust Sanger Institute. As a result the discriminatory power displayed here is rather limited in comparison and we do not attempt to resolve sample swaps as standard practice. Situations such as plate rotations would almost certainly be detected using this approach and could potentially be resolved. However due to the complex nature of trying to correlate SNPs with CNVRs it would not be appropriate to attempt sample swap resolution for patient data generated from the DDD project.

2.3.3 CNV Consensus Reference Set

All copy number variable regions (CNVRs) from each of the individual studies included in the CNV consensus were merged into separate sets of CNVEs using the previously defined method (see **Methods**). In this section we take a detailed look at various properties of each of these sets.

Size Distribution - Study Sets

An interesting and useful aspect of creating a CNV consensus from a variety of data sources is that each technology displays a different sensitivity across the genome size range. Below (see **Figure 2-19**) the size range of CNVRs included in the CNV consensus from each of the individual data sources is shown:

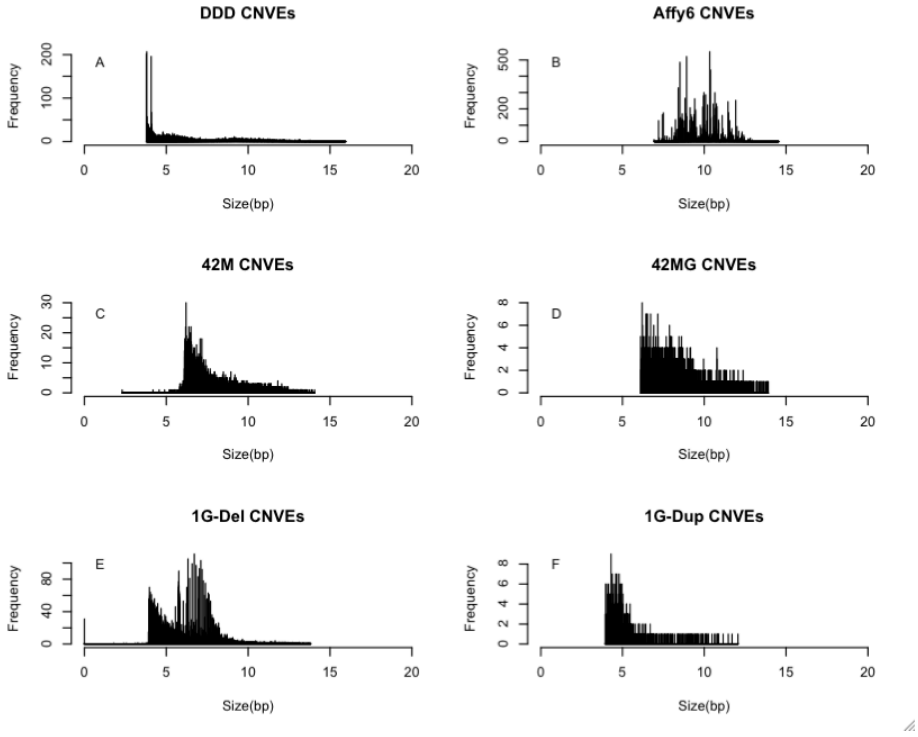


Figure 2-19 CNV Consensus - study set size distributions.

This combination of study sets achieves reasonable coverage of CNV events across the genomic size range for CNVs. The DDD arrays gives good coverage across the entire size range whereas, for example, the 1G dup set contains mostly small CNVEs and the Affy6 set contains mostly larger CNVEs. The 'dual' spike observed at the lower tail of the DDD size range is likely due to a difference in the achieved coverage of individual exons (single exon CNVs) in the DDD design.

Reciprocal Overlap Thresholds

Study CNVE Sets

Next (see **Figure 2-20),** for a number of the study sets, is plotted the proportion of CNVEs from that study set that is overlapped by each of the others, using a reciprocal overlap threshold ranging from 0 to 1.

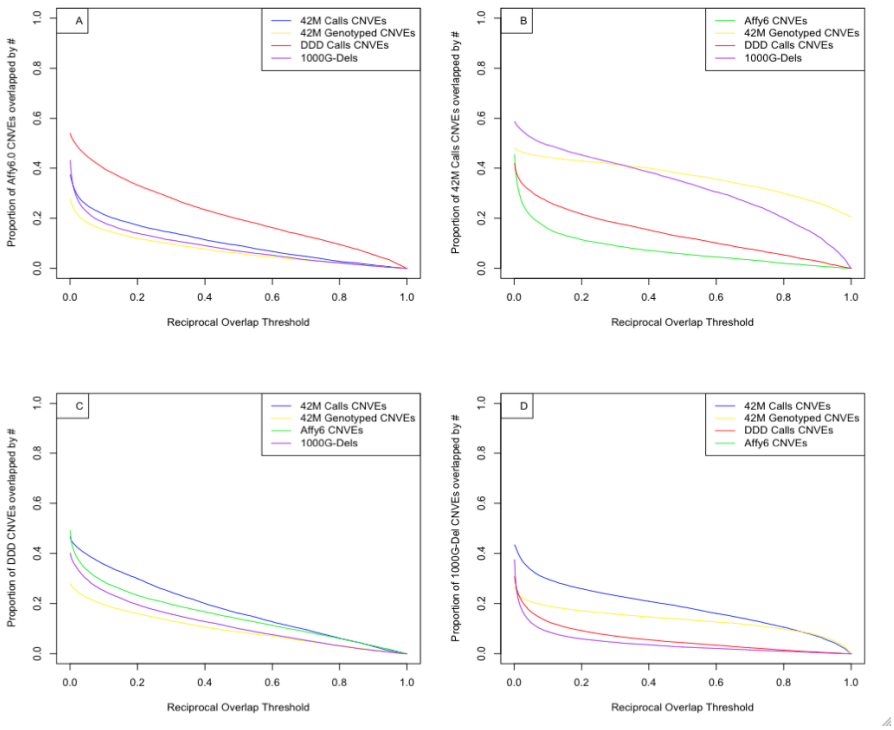


Figure 2-20 Overlap measures between study set CNVEs.

It can be observed (see **Figure 2-20**) that in all cases as the reciprocal overlap threshold tends toward 1 the proportion of CNVEs overlapped between the different study sets decreases. This is explained by the fact that each study assay is of a different resolution and as the overlap threshold gets closer to an exact match (reciprocal overlap tending toward 1) the probability of two studies sharing CNVEs is certain to decrease. Overall there is a fair amount of discordance between the 4 different study sets, showing that each is able to capture different locations of variation throughout the genome. The overlap of the affy6 CNVEs is best for the DDD control set which is likely due to the larger sample sizes of both the Affy6 and DDD control studies, 5919 & 845 respectively.

Study Call Sets

Below, (see **Figure 2-21**), similar plots are shown but comparing between study call sets instead of CNVEs where appropriate. Raw call lists were only available for the 42M, WTCCC and DDD studies.

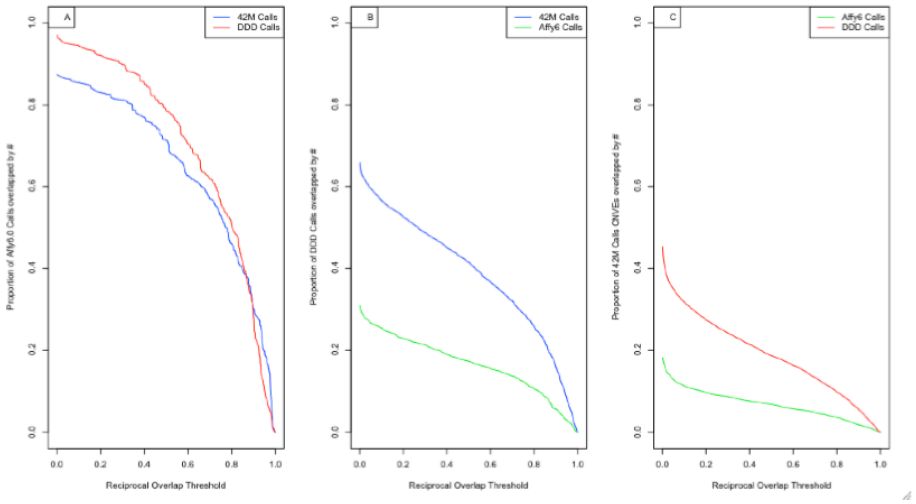


Figure 2-21 Overlaps between study set Call lists.

Here it can be observed that:

- In all cases the proportion of overlap between call lists tails off towards a reciprocal overlap of 1.
- For Affy6, the DDD call list shows a greater overlap than the 42M.
- For DDD, the 42M call list shows a greater overlap than the Affy6.
- For the 42M, the DDD call list shows a greater overlap than the Affy6.

This suggests that the DDD call set display better consistency when compared to different studies. The fact that the DDD call list shows the most similarity to both the Affy6 and 42M call lists indicates that the quality of CNV calls in the DDD list is reasonable. Furthermore, this observation indicates that the DDD array-CGH array achieves a relatively good capacity for detecting CNVs across the genomic size range. These observations are also consistent when considering the different resolutions and sample sizes of the three studies. The 42M study has the highest resolution but the lowest sample size whereas the Affy6 study has the lowest resolution but the largest sample size. The DDD controls has both high resolution and reasonable sample size and therefore is able to detect 42M detected variation that the Affy6 study was not and visa versa.

Types and Frequency Bins

WTCCC - Affy6 merged CNVEs

The Affy6 WTCCC data set (see **Figure 2-22** and **Table 2-3**) has the largest sample size of all the studies in the CNV consensus. There are a total of 5919 samples that contribute raw calls into both the deletion and duplication merged CNVE set from the WTCCC.

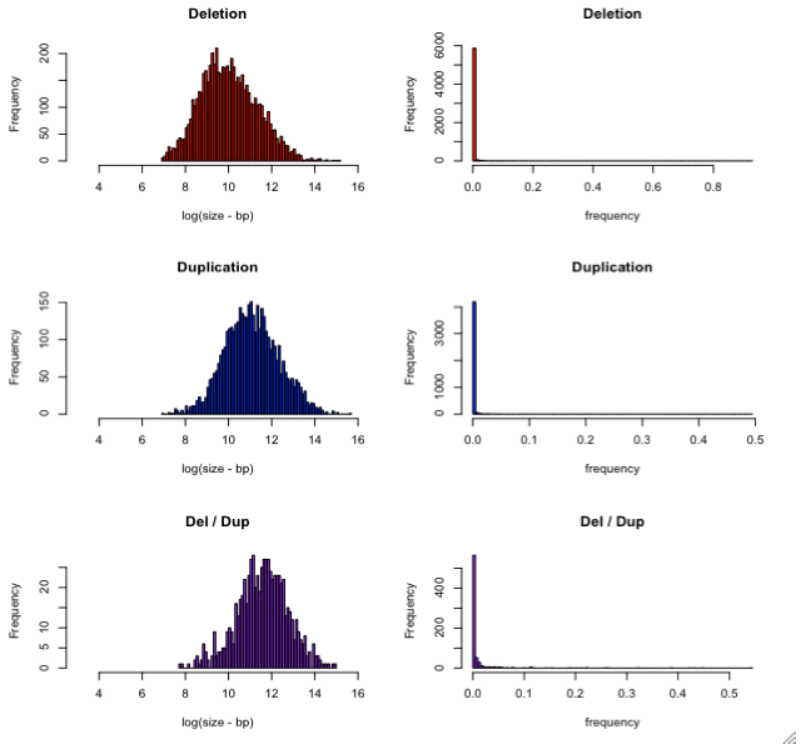


Figure 2-22 Affy 6 CNVE size (left) and frequency (right) distributions. Top - Exclusively deletion sites. Middle - Exclusively duplication sites. Bottom - Sites including both deletion and duplication CNV events.

Table 2-3 Affy6 merged CNVE set at discrete frequency bins.

	Singleton	Doubleton to 1%	1% to 5%	Above 5%
All	7103	3636	259	186
Deletion	4048	1823	127	123
Duplication	3055	1198	62	24
DelDup	0	615	70	39

Overall the Affy6 CNVE set shows relatively good separation across the frequency bins, with clearly decreasing numbers of CNVEs as the frequency bin range tends towards higher values (see **Table 2-3**). The overall singleton rate (approximately 63%) is relatively high due to the large sample size. The overall number of CNVEs is relatively low reflecting the marked decrease in CNV detection sensitivity when using SNP genotyping (Affy6) compared to array-CGH platforms.

42M - Merged Raw Calls

The 42M study (merged raw calls) has a low sample size, with a total of 40 samples contributing to the raw call list. It is however, a very high resolution study, containing 42 million probes evenly spread across the genome with a median probe spacing of 50bp (see **Figure 2-23** and **Table 2-4**).

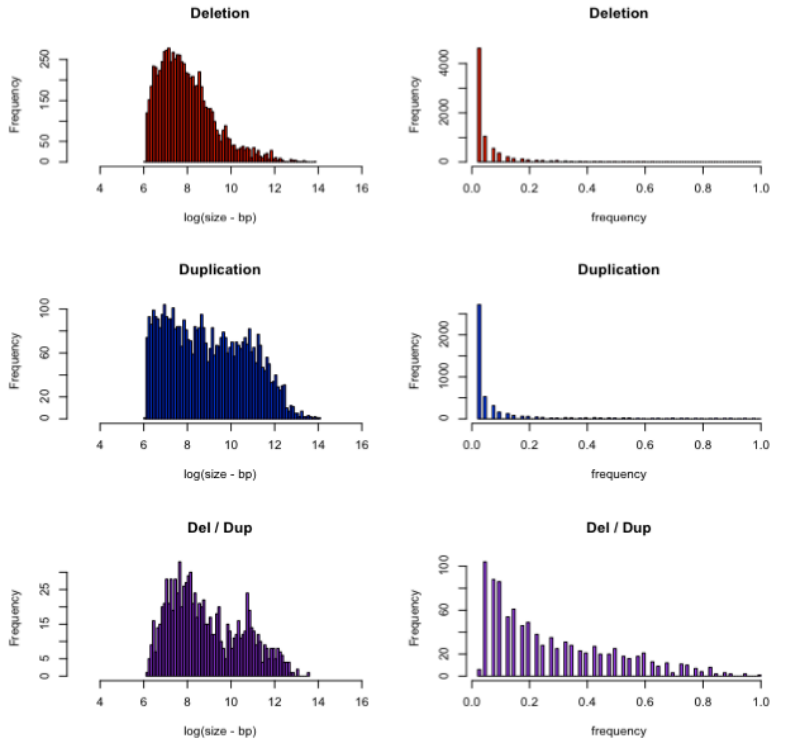


Figure 2-23 42M size (left) and frequency (right) distributions. Top - Exclusively deletion sites. Middle - Exclusively duplication sites. Bottom - Sites including both deletion and duplication CNV calls.

Table 2-4 42M merged CNVE set at discrete frequency bins.

	Singleton	Doubleton to 1%	1% to 5%	Above 5%
All	7351	0	9037	4121
Deletion	4630	0	5682	1944
Duplication	2721	0	3245	1312
DelDup	0	0	110	865

The clear observation that the merged CNVE set from the 42M study has no events within the doubleton to 1% frequency (see **Table 2-4**) range is a consequence of its small sample size (40 samples). This is explained by the fact that a CNV observed in two independent samples occurs at a 5% frequency in the 42M study. Even though the sample size is small the singleton rate overall is relatively low (approximately 36%) and the overall number of CNVEs is relatively high suggesting that the CNV call quality and detection sensitively was relatively good in the 42M study.

DDD - Merged Raw Calls

The DDD arrays contribute both high resolution and a reasonable sample size into the CNV consensus. The array provides a good compromise between backbone resolution and targeted genomic element coverage (see **Figure 2-24** and **Table 2-5**).

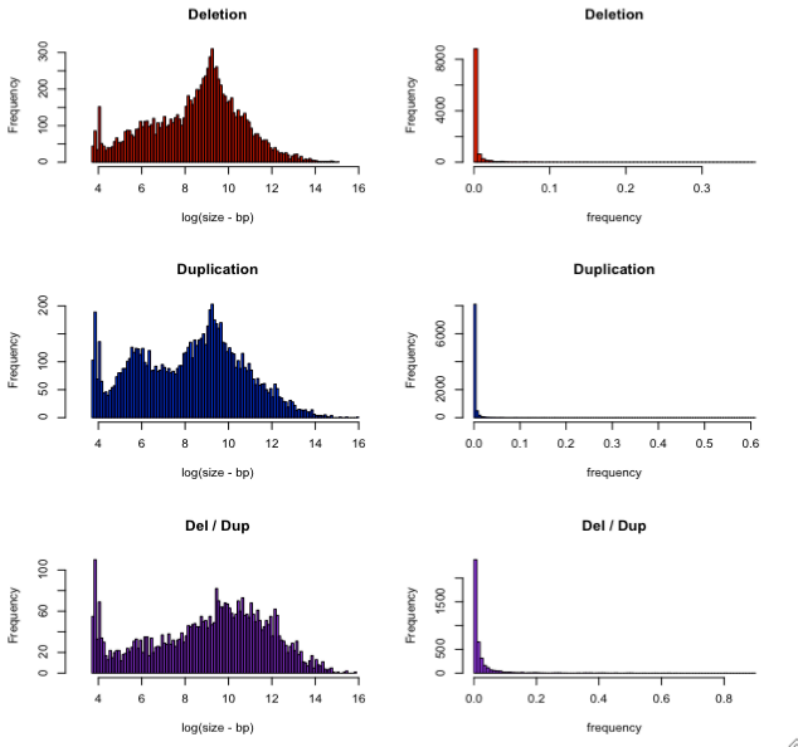


Figure 2-24 DDD size (left) and frequency (right) distributions. Top - Exclusively deletion sites. Middle - Exclusively duplication sites. Bottom - Sites including both deletion and duplication CNV calls.

Table 2-5 DDD merged CNVE set at discrete frequency bins.

	Singleton	Doubleton to 1%	1% to 5%	Above 5%
All	11789	7916	2362	671
Deletion	6138	2949	757	179
Duplication	5651	2579	361	57
DelDup	0	1679	1244	435

Overall the DDD CNVE set shows good coverage across the genomic size range (see **Figure 2-24**). Furthermore, due to the relatively high sample size the separation across the frequency bins shows good separation and the expected decrease as the frequency range tends towards one (see **Table 2-5**). The singleton rate is relatively low (approximately 52%) and the overall number of CNVEs is high, indicating good CNV call quality and detection sensitivity in the DDD control samples.

DDD Controls in the CNV Consensus

CNV Consensus version1 - without DDD

The CNV consensus set version1 contains all study sets excluding the DDD control data set. Below (see **Figure 2-25**) the size (left) and frequency (right) distribution of CNVEs contained inside the CNV consensus version1 is shown.

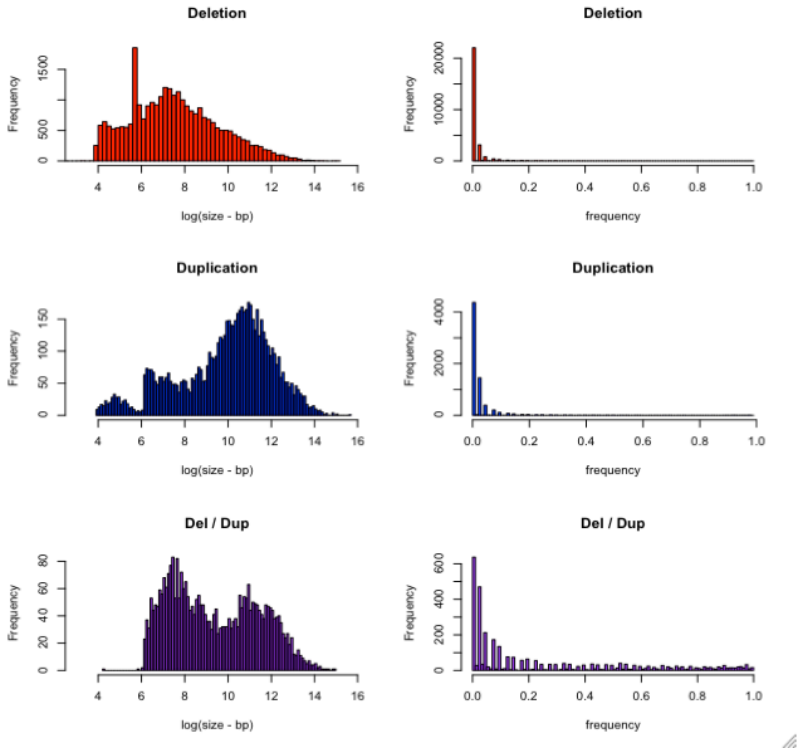


Figure 2-25 CNV consensus v1 size (left) and frequency (right) distributions. Top - Exclusively deletion sites. Middle - Exclusively duplication sites. Bottom - Sites including both deletion and duplication CNV calls.

Table 2-6 CNV consensus version 1 CNVE set at discrete frequency bins.

	Singleton	Doubleton to 1%	1% to 5%	Above 5%
All	15327	7251	6084	5833
Deletion	10634	5991	5240	4368
Duplication	4689	889	356	600
DelDup	0	417	226	1081

Overall the combined CNV consensus reference set achieves good coverage across the genomic range without the DDD CNVE set (see **Figure 2-25**). The overall number of CNVEs shows a marked increase compared to individual studies and the singleton rate (approximately 44%) is between that of the Affy6 (large sample size) and 42M (small sample size) studies, suggesting a reasonable refinement of the positional and the frequency based estimates for CNVEs when combining information across studies (see **Tables 2-3, 2-4, 2-5 & 2-6**).

CNV Consensus version2 - with DDD

Below (see **Figure 2-26**) shows the same plots as in **Figure 2-25** except that the DDD controls sets have been added into the CNV consensus.

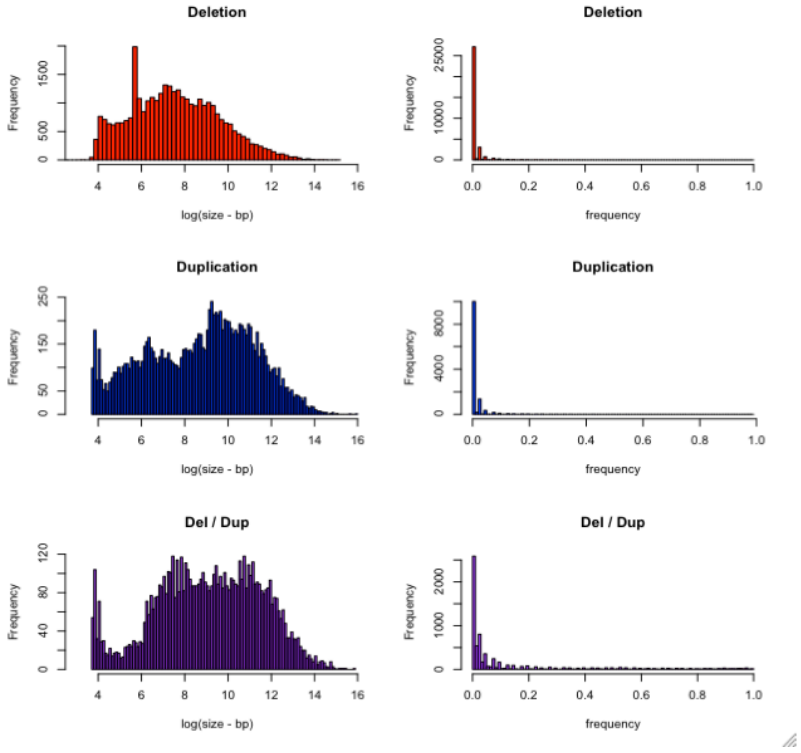


Figure 2-26 CNV consensus v2 size (left) and frequency (right) distributions. Top - Exclusively deletion sites. Middle - Exclusively duplication sites. Bottom - Sites including both deletion and duplication CNV calls.

Table 2-7 CNV consensus version 2 CNVE set at discrete frequency bins.

	Singleton	Doubleton to 1%	1% to 5%	Above 5%
All	23390	12698	7894	6862
Deletion	14665	8085	5857	4692
Duplication	8718	2840	640	688
DelDup	0	1722	1128	1802

By integrating the DDD control CNVEs into the CNV consensus the number of CNV locations has increased by approximately 32% (see **Table 2-7**). However the overall singleton rate has only been increased by 2%, indicating that by incorporating the CNV calls made by CNsolidate across the DDD control samples the CNV consensus has a significant increase in the number of CNVE locations with more than one observation across studies (not singleton).

There are 5,448 merged genomic locations in the DDD merged CNVE set that do not show any overlap (at least 1 base pair) with the version 1 of the CNV consensus. The singleton rate within these 5,448 CNVEs totally unique to DDD study is approximately 42%, a similar rate to the overall combined CNV

consensus with and without the DDD CNVE set. Below (see **Table 2-8**) the numbers in the different frequency bins of the DDD CNVEs that had no overlap with any CNVE from version 1 of the CNV consensus is presented:

Table 2-8 DDD CNVEs - no overlap with CNV consensus version 1 at discrete frequency bins.

	Singleton	Doubleton to 1%	1% to 5%	Above 5%
All	3182	1468	558	240
Deletion	1615	555	159	45
Duplication	1567	561	82	22
DelDup	0	354	316	172

The overall characteristics of the 5,448 merged CNVE locations unique to the DDD CNVE set (see **Table 2-8**) are encouraging. The separation between the frequency bins shows good separation and the expected decrease in numbers as the frequency range tends towards one. The number of exclusively deletion and duplication events shows a ratio of approximately one, suggesting that the sensitivity for detecting CNV regions unique to the DDD control samples and array design was near equivalent for deletions and duplications using CNsoliate.

The combination of these characteristics give a good indication that the DDD raw call list was made up of high quality CNV detections. For example, the number of singletons observations in a control set of common variation can act as a good sanity check, where studies display unusually high singleton rates (CNVEs that were observed in single samples) this can be an indication that the study under question may have higher than desired false positive rate.

Combined CNV Consensus Overview

The full combined CNV consensus set is composed of a set of non-overlapping CNVEs that were merged across all studies and contain combined positional information, frequency estimates and type states (see **Methods**). Additionally each CNVE retains a 'study tag' that relates to the number and type of studies that contributed information to the positional, frequency and type definitions of the CNVE. Below (see **Figure 2-27**) is a summary of the CNVEs contained inside the full CNV consensus set.

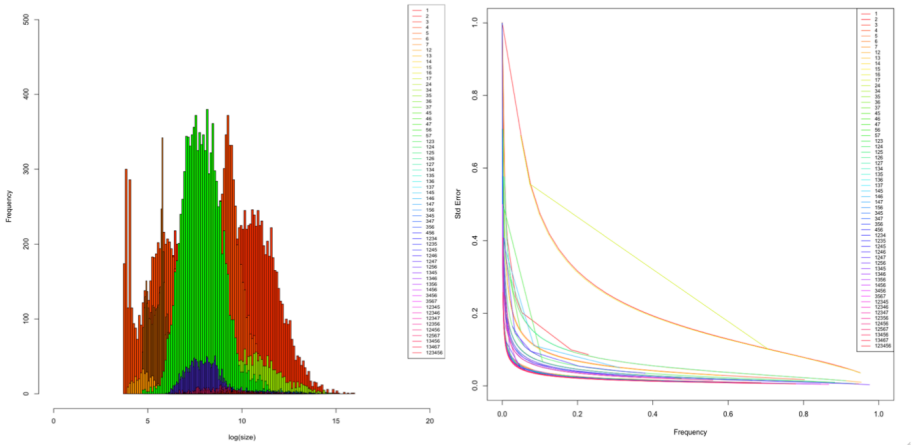


Figure 2-27 Left - Size distribution of CNVEs in the combined CNV consensus set Right - Standard error of CNVEs in the combined CNV consensus set.

In both panels, the colour scale bar on the right denotes the specific study tag (which studies contribute to a particular CNVE). The left panel displays the size distribution of all CNVEs, it is clear to see that each study tag contributions CNVEs displaying a different range of genomic sizes.

Overall, the combined CNV consensus set achieves good coverage across the entire genomic range for CNVs. Sequencing based studies, such as the 1000G data sets, contribute mostly smaller CNVEs, whereas the microarray based studies, such as the 42M data sets, contribute a range of CNVEs including the typical larger structural rearrangements (see **Figure 2-27**). The right panel displays the standard error associated with each CNVE. This varies between study tags due to differences in the sample sizes of individual study sets and the number of observations made at each CNVE (see **Figure 2-27**).

CNV Consensus Set Visualisation

The CNV consensus set can be made available to anyone who has a need to accurately filter CNV data based on positional, frequency and type information. The CNV consensus set is in use for CNV filtering in a number of different projects based at the Wellcome Trust Sanger Institute and beyond. Furthermore, the positional, frequency and type information can be visualised using online web based resources such as DECIPHER [16], ENSEMBL [163-166] and UCSC [167]. For example below (see **Figure 2-28**) is a screen shot of a section of chromosome1 from the CNV consensus displayed in the UCSC genome browser.

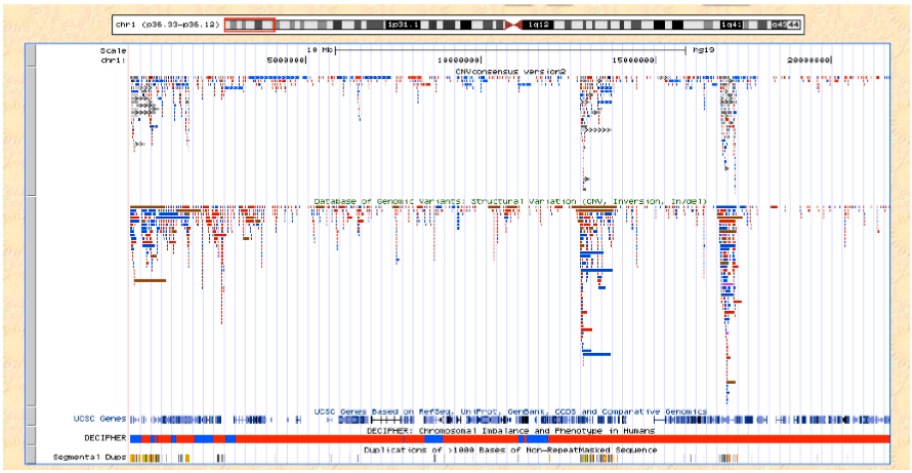


Figure 2-28 UCSC Screen Shot of the CNV Consensus Reference Set.

The CNV consensus set is displayed as the top track with the Database of Genomic Variants (DGV) [168] track directly underneath it (see **Figure 2-28**). The colour of the bars in the CNV consensus track denotes the CNVEs type set and the shade denotes the frequency estimate. The two tracks, CNV consensus and DGV show a fair degree of similarity, especially in complex regions of the genome containing large numbers of segmental duplications (bottom track). This is expected as genomic regions flanked by segmental duplications tend to contain large numbers of common copy number variation due to CNV formation mechanisms such as Non-allelic homologous recombination. The major difference between the CNV consensus set and the DGV is the fact that the consensus set incorporates information from only a small number of carefully selected studies and additionally provides an estimate of population frequency and state type for each of its CNVEs.

Defining Common, Rare and Novel CNVs using the CNV Consensus Set

For annotating population frequency information onto CNV calls made from patient data in the DDD project we choose to define three frequency bins, common, rare and novel. All patient CNV calls in the DDD project are annotated with one of these frequency bins by applying a number of overlap and frequency rules during comparison to the CNV Consensus reference set.

Parameters

The CNV Consensus is split into two sets (common and rare) based on the observed population frequency using approach 1 (see **Methods**). Common CNVEs are defined as those observed at greater than or equal to, and rare CNVEs as those observed at less than, 1% population frequency from within the CNV consensus set. All 'test' (patient) CNV calls are compared against both frequency sets (common and rare) by CNV type (deletion or duplication), meaning that test deletion calls are only compared against CNVEs observed as only deletion, or deletion/duplication (complex) events and test duplication calls are only compared against CNVEs observed as only duplication, or deletion/duplication (complex) events from the CNV consensus. For every test CNV two overlap measures are calculated, maximum forward overlap and maximum backward overlap, where the maximum forward overlap is the maximum proportion of the

test CNV call overlapped by a CNVE of the given frequency bin and CNV type and the maximum backward overlap is the proportion of the given CNVE overlapping the test CNV call (see **Figure 2-29**).

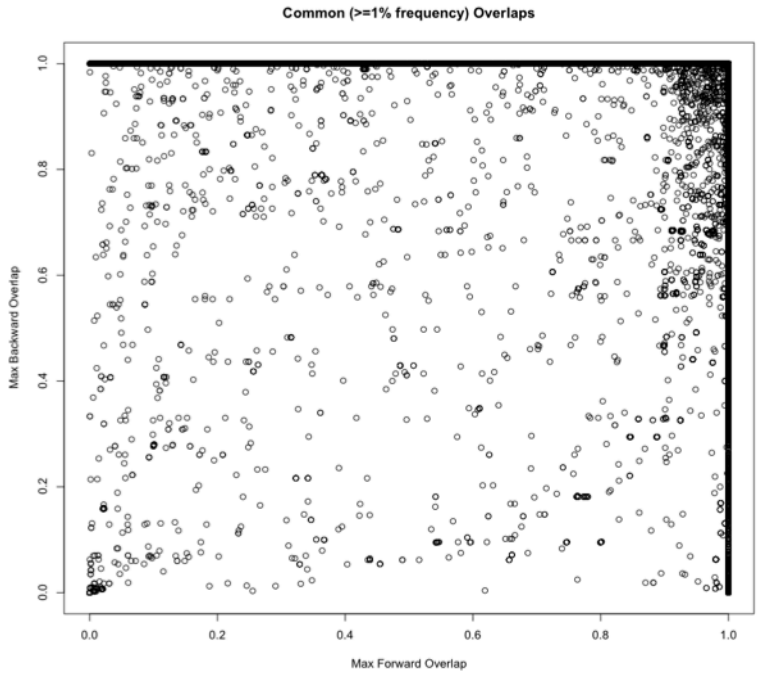


Figure 2-29 The common forward and backward overlap measures calculated when comparing CNV calls from 204 DDD patient samples to the CNV Consensus.

Above (see **Figure 2-29**) is a plot showing the forward and backward overlap measures for CNV calls in 204 DDD patient samples compared to the common (>1% population frequency) CNVEs from the CNV consensus reference set. To define test (patient) CNV calls as common cut-offs on both the forward and backward overlap measures needed to be defined. Cleary test CNV calls with both a forward and backward overlap measure of one should be classed as common since the breakpoint locations of the test CNV are exactly the same as the common CNVE from the CNV consensus set. Conversely, test CNV calls with both a forward and backward overlap measure of zero should not be classed as common since they share no overlap with a common CNVE of the given CNV type from the CNV consensus. However exactly where to define the cut-offs is somewhat arbitrary and is highly likely to be influenced by the resolution achieved by the array used for test CNV calling and the resolution of the studies included in the reference set. As the CNV consensus reference set containing a number of studies and achieves good coverage of the genomic size range (see **Figure 2-27**) we choose to use a relatively strict cut-off for defining CNV states (common, rare and novel).

We use a greater than 0.8 forward overlap for defining common and rare CNV calls where 'test' CNV calls with a forward overlap measure of greater than 0.8 for common and rare CNVEs of the given type from the CNV consensus reference

set are defined as common and rare respectively. We define test CNV calls as novel when neither the "common forward overlap" or the "rare forward overlap" measure are greater than 0.8. We choose to ignore the backward overlap during these definitions and assume high enough similarity so long as greater than 80% (forward overlap > 0.8) of the 'test' CNV call is encompassed by the reference CNVE.

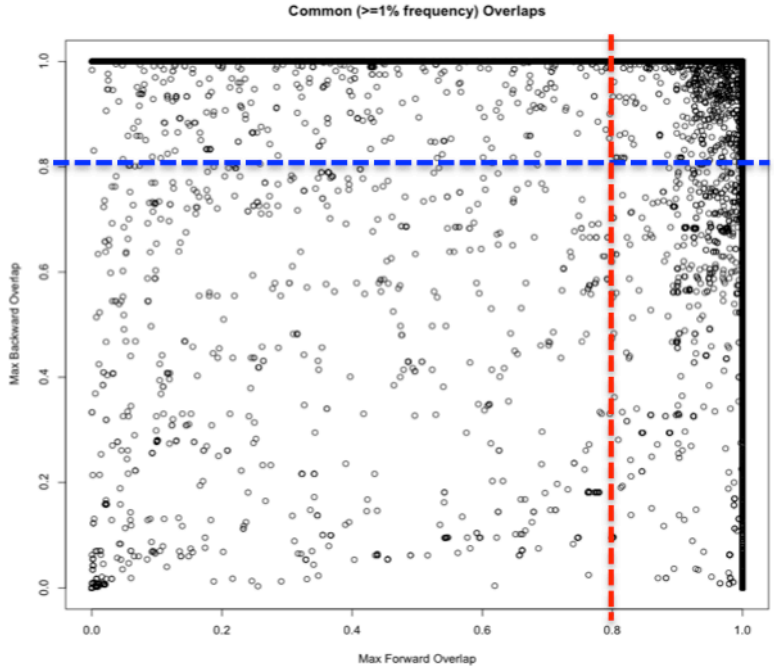


Figure 2-30 Highlighting the overlap cut-off definitions for defining common, rare and novel CNV states using the CNV consensus reference set.

The above plot (see **Figure 2-30**) highlights the defined cut-off on positional similarity for defining CNV frequency states. The red line shows the operational cut-off used on the forward overlap measure. The blue line shows the position on the graph if we were to use the same cut-off (0.8) on the backward overlap measure. Choosing to ignore the backward overlap measure for defining CNV frequency states is almost certainly an over simplification as CNVs encompassing different areas of the genome are likely to confer different phenotypic effects and should not strictly be classed as the same event. For example, a small test CNV call affecting a specific genomic location (e.g. a single exon) may result in an altered function of the gene product compared to a lowered or increased gene dosage conferred by either a deletion or duplication of an entire gene(s). Overall the positional effects of CNVs inside, outside or including genes are generally poorly understood. However, given the previously mentioned problems associated with different assay resolutions, using a strict greater than 0.8 cut-off on the forward overlap measure alone is a reasonable compromise for assigning a population frequency estimate to individual CNV calls.

2.3.4 CNV Filtering

Sample Set

We applied CNV detection using CNsolidate, CNV frequency state definitions using the CNV consensus reference set and CNV filtering for clinical relevance to 1288 QC passed proband (patient) array-CGH datasets from the DDD project.

Variant Description

The 1288 array-CGH datasets all passed quality control (data quality and data tracking) given the previous described methods and parameter definitions (see **Methods**). They were therefore eligible for further analysis and ultimately reporting of clinically relevant variants using the DECIPHER database back to the clinical services throughout the UK and Ireland.

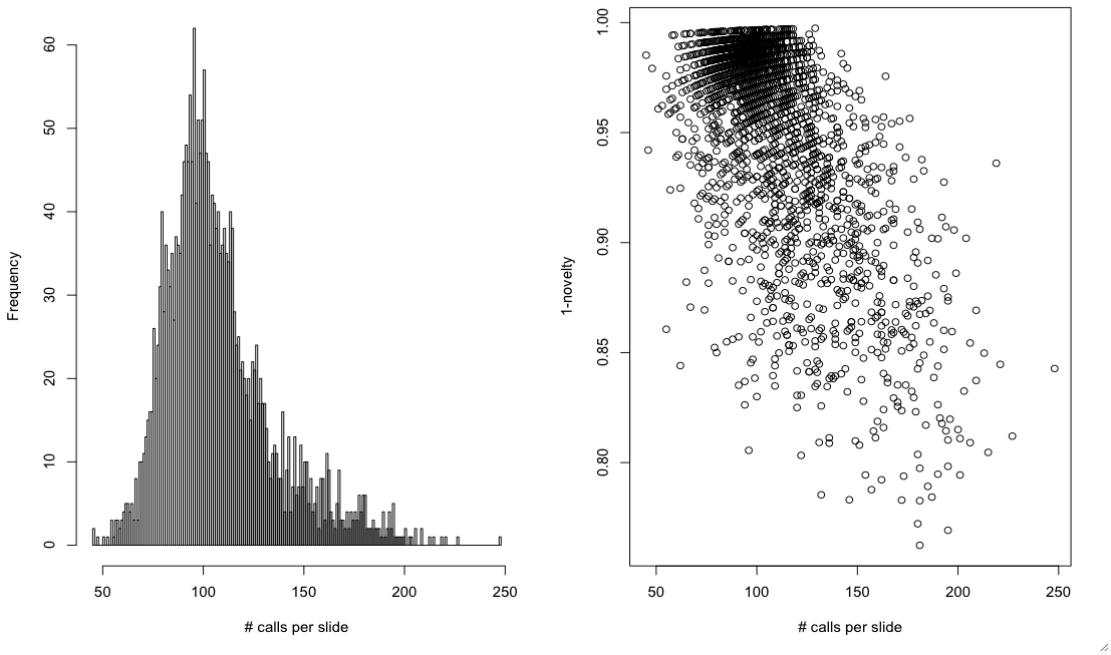


Figure 2-31 Left – the number of QC passed CNV calls per array-CGH slide, Right – the number of QC passed CNV calls per array-CGH slide vs. 1-novelty (the proportion of CNV calls previously observed in the CNV consensus reference).

Overall the QC parameters used for passing individual samples based on data quality used were relatively relaxed. This is because it was important to maximize the number of patients within the DDD study who could potentially be provided with a diagnosis. Furthermore every flagged variant is manually reviewed by a panel of experts prior to being deposited in DECIPHER. Even so both the number of CNV calls per slide and the 1-novelty values are relatively consistent across all 1288 samples (see **Figure 2-31**).

CNVs Flagged for Clinical Review

Overall 382 unique CNVs were flagged from a total of 232/1288 individual samples, a potential feedback rate of 18%. There was a median of 0, mean of 0.31 and maximum of 10 CNVs flagged per sample (see **Figure 2-32**), reflecting the relatively high stringency applied for call quality control (see **Methods**). Furthermore greater than 80% of samples had a previously uninformative microarray result from the clinical services and front line microarray diagnostic testing normally results in less than a 20% diagnostic rate [169].

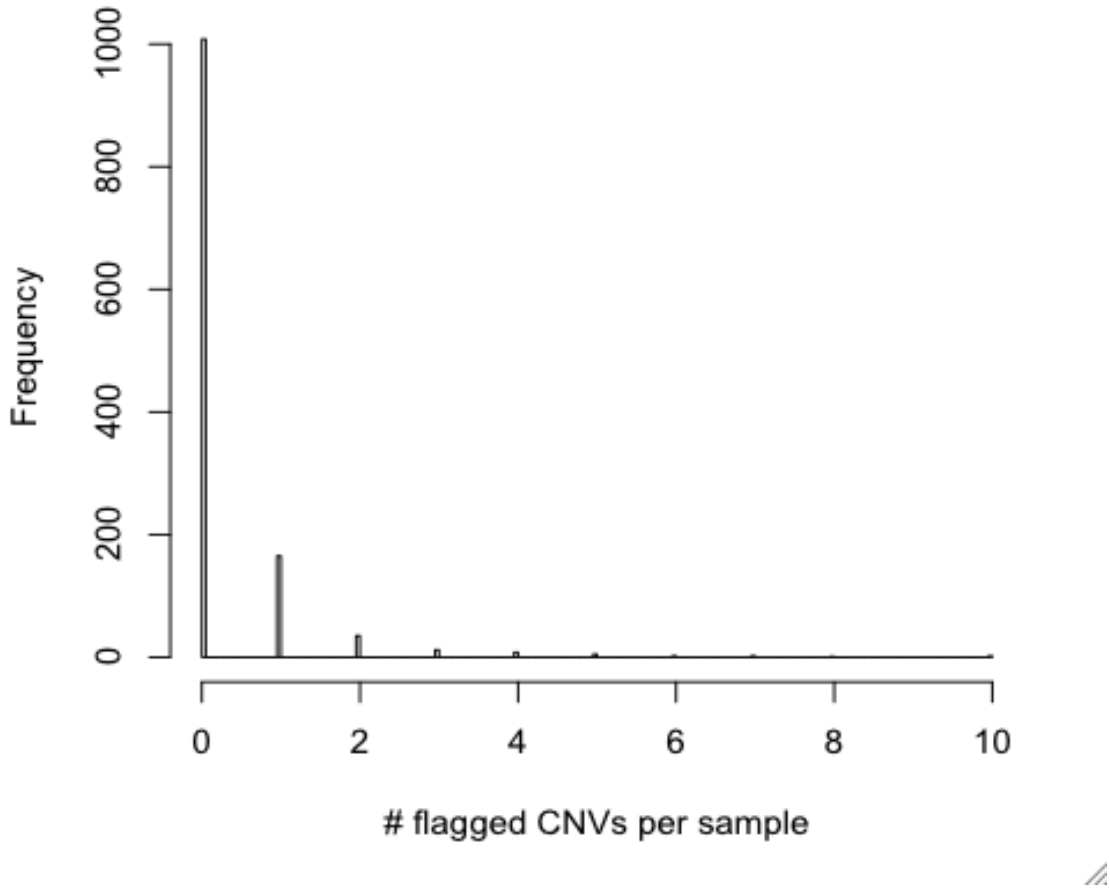


Figure 2-32 the number of CNVs flagged for clinical review per sample.

As shown (see **Figure 2-32**) the majority of samples had zero CNVs flagged for clinical review and across the samples with at least one flagged CNV there was a median of 1, mean of 1.6 and maximum of 10 flagged CNVs. This shows that the parameters used for clinical filtering were optimal in terms of providing relativity low numbers of flagged CNVs per sample. Although this relates to a potentially low feedback rate it ensures a low burden on the manual review stage. Furthermore, the CNV filtering pipeline has been designed to be iterative in nature and further rounds of variant filtering and review can happen over the course of the DDD project where filtering parameters should be adjusted.

Flagged Variant Breakdown by Filtering Route

As described previously the CNVs flagged for clinical review can come down one of two different routes, either flagged based on size and frequency alone (VOUS) or due to affecting a gene previously associated with developmental disorders (DDG2P). Overall 165 (43%) of flagged CNVs came down the VOUS route and were flagged based on size and frequency alone whereas 217 (57%) came down the DD gene route and were flagged due to overlapping a DD gene with the correct mode and mechanism (see **Methods**). Overall there were 27 homozygous deletions, 266 heterozygous deletions and 89 gains flagged across both the DD gene and VOUS filtering routes.

Table 2-9 The number of CNVs of copy number 0, 1 and greater than 2 flagged for clinical review via either the DD gene or VOUS filtering route

	Copy number 0	Copy Number 1	Copy Number > 2
DD Gene Route	25	187	5
VOUS Route	2	79	84

Reviewing the numbers displayed (see **Table 2-9**) it was clear that for homozygous deletions (copy number = 0) there was a greater than ten-fold increase in the number of variants flagged via the DD gene route compared to the VOUS route. This was not unexpected as the rate of large rare homozygous deletions is very low across the genome [170] and these types of events result in zero copies of all effected genes and therefore have a high prior likelihood of causing extreme phenotypes [171-174].

There was no observable bias between the number of losses and gains flagged via the VOUS route indicating that the capacity for detecting and prioritizing large rare CNVs for clinical review was near equivalent for losses and gains. However for the DD gene route there was a significant enrichment for losses compared to gains (fisher **P< 2.2e-16**), this enrichment is almost certainly dominated by the bias for loss of function genes within the DDG2P database; with 72% of entries having loss of function or dominant negative mechanisms and only 3.5% having either increased gene dosage or activating mechanisms. Furthermore the loss of one or more functional gene copy is clearly more likely to result in a genetic disorder than simply an increase in the number of functional gene copies. Therefore unless the gene transcript is interrupted or truncated by a gain in copy number, resulting in impaired gene product function (disruptive duplications) we can expect a greater number of genetic disorders caused by CNVs to be a result of a loss compared to a gain of genetic material.

The size distribution of CNVs flagged via the DD gene or VOUS route are understandably very different due to fixed parameters on CNV length for variants flagged via the VOUS route. For the DD gene route there was a minimum of 44 bp, a median of 7.8 Kb and a maximum of 77 kb whereas for the VOUS route there was a minimum of 103 kb, a median of 713 Kb and a maximum of 14.6 mb across all flagged CNV sizes.

Recurrently Mutated DD Genes

The specific version of the DDG2P gene list (version 1.2) used for this run of clinical filtering contained a total of 1,204 unique entries with 1,055 discrete genes. All 217 CNVs flagged via the DD gene route contained a total of only 35 discrete DD genes indicating a high rate of recurrently mutated DD genes across the 1,288 samples.

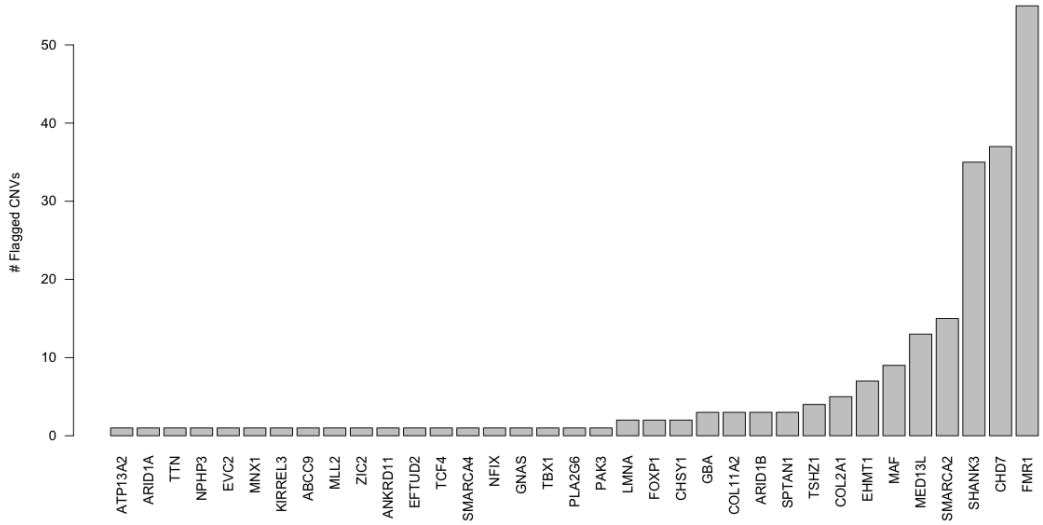


Figure 2-33 Number of CNVs flagged via the DD gene route per DD gene

Overall the number of CNVs observed per DD gene shows three clear outliers (see **Figure 2-33**) in terms of the number of times they were hit by single CNVs (FMR1, SHANK3 and CHD7). Otherwise six genes with greater than three CNVs were observed and a long tail with three or fewer flagged CNVs. The three genes (FMR1, SHANK3 and CHD7) with the greatest number of flagged CNVs indicated a potential problem with the filtering pipeline due to being observed at a high enough frequency (4%, 2.8% and 2.7% respectively) across the 1,288 sample to arise suspicion.

Furthermore during the clinical review meeting we had observed variants at these three genes that were high quality CNV calls however the phenotypes listed for the patients were often irrelevant and not associated with the described genomic syndrome for the particular DD gene.

As a result we concluded that a number of probes on the array-CGH array were potentially poorly performing and appearing as deletion calls in a subset of the patient data. This effect was not observed in control data and CNVs at FMR1, SHANK3 and CHD7 were not present in the CNV consensus set. Presumably this effect was observed due to differences in DNA sources and quality for the patient compared to the control samples. The patient DNA was obtained as either extracted blood or saliva DNA from a large variety of different sources whereas the control data was generated from cell line extracted DNA.

Defining Problem Probes for Removal Prior to CNV Calling

Due to the observations described above an association-based test was undertaken for every probe present of the array-CGH platform across the 1,288 samples. We defined two different sample sets to compare:

- Sample set 1:
    - 129 samples: containing a deletion call at either SHANK3, CHD7 or FMR1.
- Sample set 2:
    - 1,159 samples: containing no deletion call at either SHANK3, CHD7 or FMR1.

Two different measures of probe performance for each probe (1.92 Million for each sample) between sample set 1 and sample set 2 were calculated:

1) The absolute difference between the mean log2 ratios between sets
2) A Mann-Whitney test between all probe values between sets

Overall there were 1,288 datasets containing 1,915,129 array-CGH probes each, equally a total of 2,466,686,152 data points to compare. To achieve this analysis in a timely fashion we made use of the load sharing facility (LSF) available on the compute farm at the Welcome Trust Sanger Institute. There was a balance needed between the number of executed jobs, the number of file reads and the memory requirements for each job.

Table 2-10 Three different approaches to running a probe association test across 1288 sample for 1.9 Million probes on a compute cluster.

	# Jobs	**# File Reads**	**Memory**
1 job per probe	1915129	1288	low
1 job per chromosome	24	102778590	low
1 job per chromosome (chunked)	24	128800	moderate

The third option displayed above (see **Table 2-10**) was used to perform the probe association test across the 1,288 samples. Using this approach one job per chromosome (24 jobs) was executed and each chromosome was split into 100 chunks. For each of the 100 chromosome chunks intensity data for all probes within the chunk across all 1,288 samples were read into memory and the probe performance measures were calculated before outputting an indexed output file containing the results.

This approach only required a moderate amount of memory and the number of file reads was also moderate (128,800) being far below the limit on the number of file reads allowed per job (1 Million) running under LSF at the WTSI. However,

each job still required 128,800 sequential file reads extracting a median of 805 data points per file read (a median of 1,036,840 total data points per chunk) meaning that each job could be potentially quite slow. For example, assuming a 0.1 second per file read each job would have taken a total of 3.6 hours to complete. However by using the fast data indexing C++ package (Rbin) all jobs completed in a total of 42 minutes (including job scheduling wait time).

Probe Association Results

Using the approach above we obtained two performance measures for every probe on the array-CGH platform.

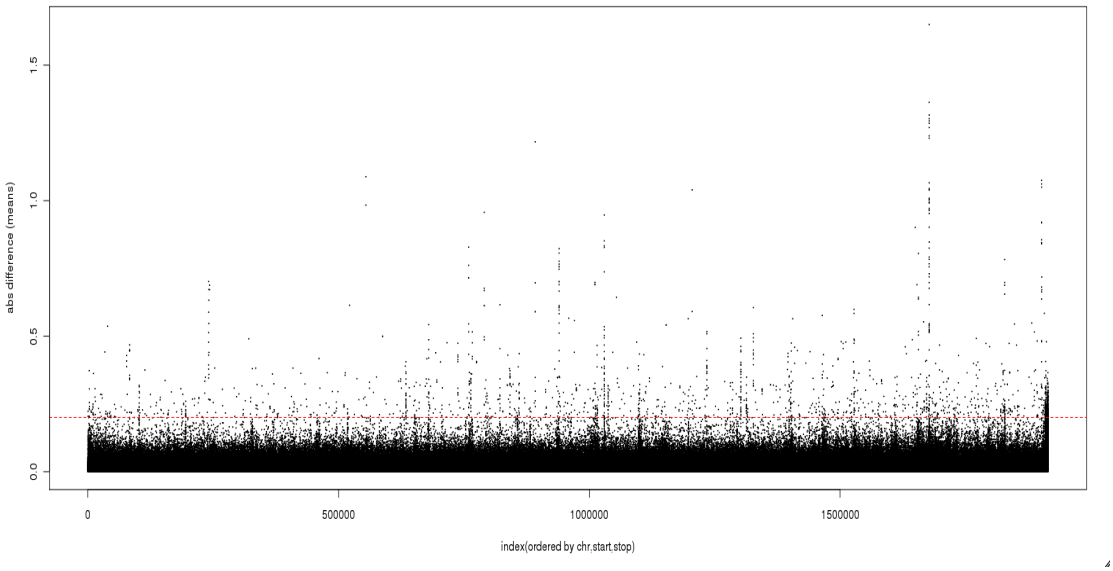


Figure 2-34 Absolute mean difference between the mean log2 ratios of sample set 1 and sample set 2 for 1.92 Million array-CGH probes, ordered by genomic location

Above (see **Figure 2-34**) the first performance measure calculated for every probe on the array-CGH platform is shown. By setting an arbitrary cut-off of greater than 0.2 on the absolute difference between log2 ratio values from sample set 1 compared to sample set 2 (excluding the problematic chromosome Y) we would remove a total of 1,535 probes (0.08% of total probes) that show a marked difference in mean log2 ratio values between the two sample sets.

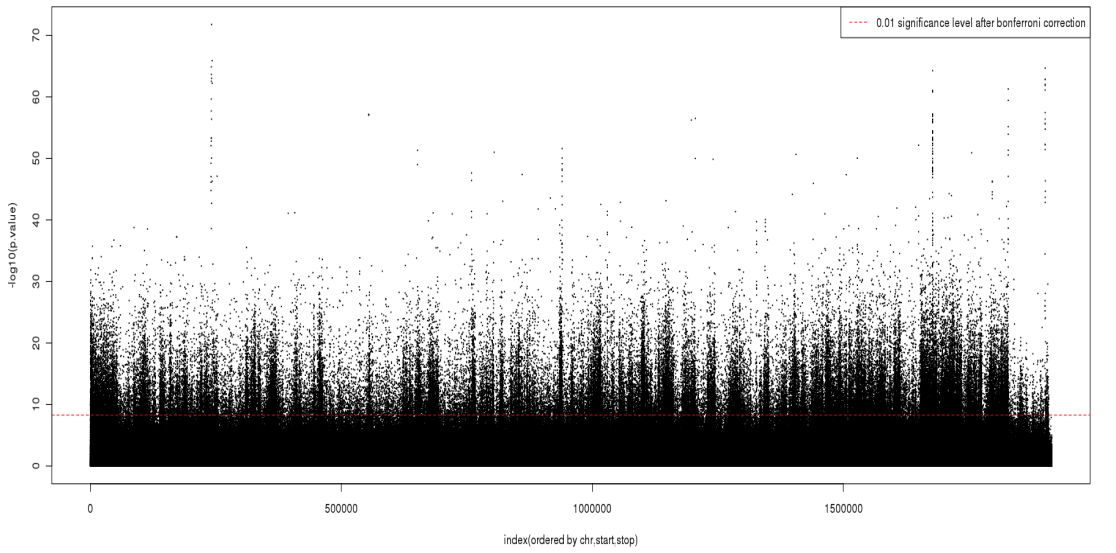


Figure 2-35 -log10(P) from a Mann-Whitney test between all probe log2 ratio values for sample set 1 compared to sample set 2 for 1.92 Million array-CGH probes, ordered by genomic location.

Above (see **Figure 2-35**) the results when using the more sensitive Mann-Whitney test to assess differences in probe performance between the two sample sets is shown. After applying a multiple testing correction (Bonferroni) at the 0.01 significance threshold we observed 90,048 probes (4.7% of total probes) with a significant different between the two sample sets.

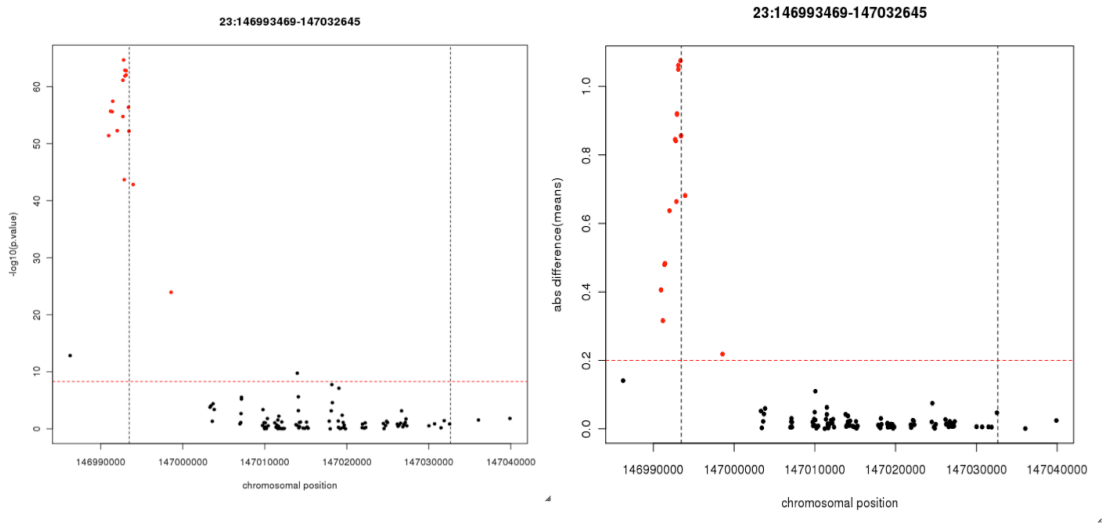


Figure 2-36 Probe performance measure for all probes covering the FMR1 gene. Left: -log10(p). Right: Abs mean ratio difference. Red dashed line: defined performance measure cut-offs. Red dots: probes with an additional Kolmogorov-Smirnov test value of zero.

Across the FMR1 gene locus there was a clear cluster of probes near the 5' end that showed significant differences in performance between the two sample sets based on both probe performance measures assessed (see **Figure 2-36**).

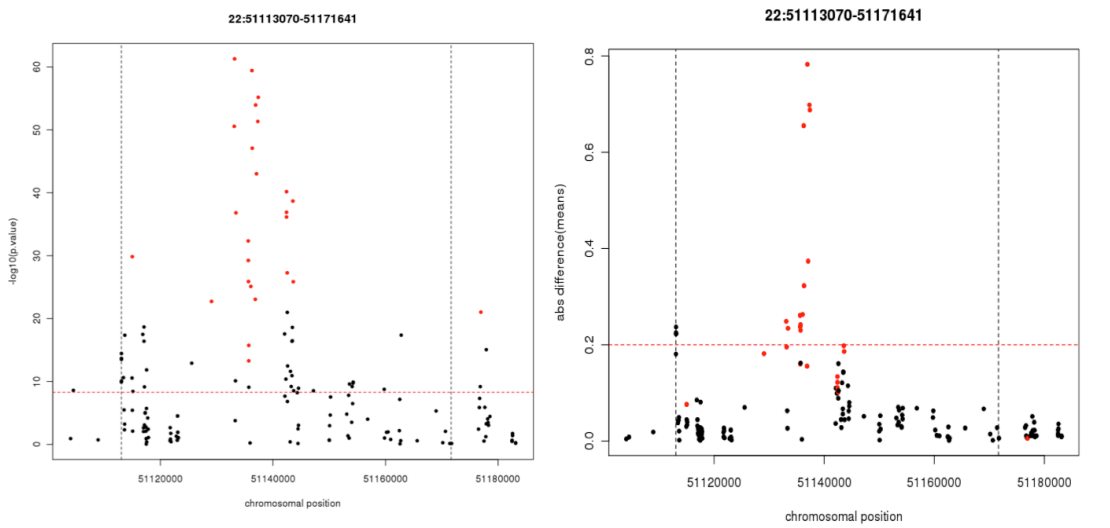


Figure 2-37 Probe performance measures for all probes covering the SHANK3 gene. Left: -log10(p). Right: Abs mean ratio difference. Red dashed line: defined performance measure cut-offs. Red dots: probes with an additional Kolmogorov-Smirnov test value of zero.

There were a number of probe clusters across the entire SHANK3 gene showing significant differences in performance between the two sample sets based on both probe performance measures assessed (see **Figure 2-37**).

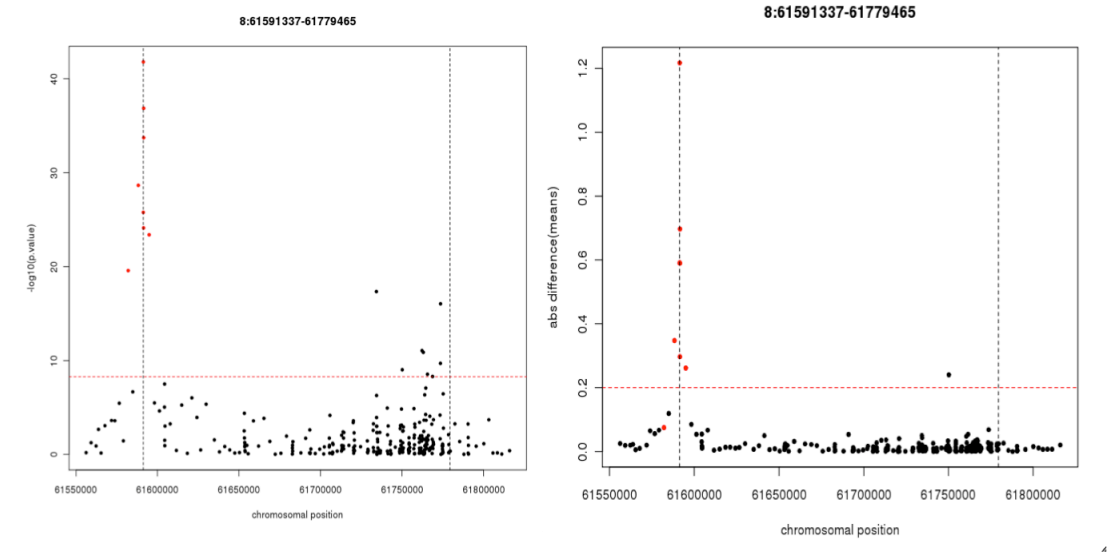


Figure 2-38 Probe performance measure for all probes covering the CHD7 gene. Left: -log10(p). Right: Abs mean ratio difference. Red dashed line: defined performance measure cut-offs. Red dots: probes with an additional Kolmogorov-Smirnov test value of zero.

Across the CHD7 gene locus the was a clear cluster of probes near the 5' end and a small cluster towards the 3' end that showed significant differences in

performance between the two sample sets based on both probe performance measures assessed (see **Figure 2-38**).

Probe Removal Results

The distributions of the two probe performance measures assessed at the three problematic genes (FMR1, SHANK3 & CHD7) displayed a high degree of visual correlation. The Mann-Whitney test p values were more sensitive to differences in probe performance between the two sample sets and a greater number of probes fall below the defined cut-off. For the probes showing the most significant differences between sample set1 and sample set2 both performance measure cut-offs resulted in their removal (all probes removed based on the absolute difference between mean log2 ratios were also removed based on the Mann-Whitney p values).

Overall we decided to remove probes based on the Mann-Whitney p values and defined a 5.22158e-09 P value cut-off relating to a 0.01 significance values after correcting for 1,915,129 independent tests. This resulted in removing 4.7% of the total number of probes on the array.

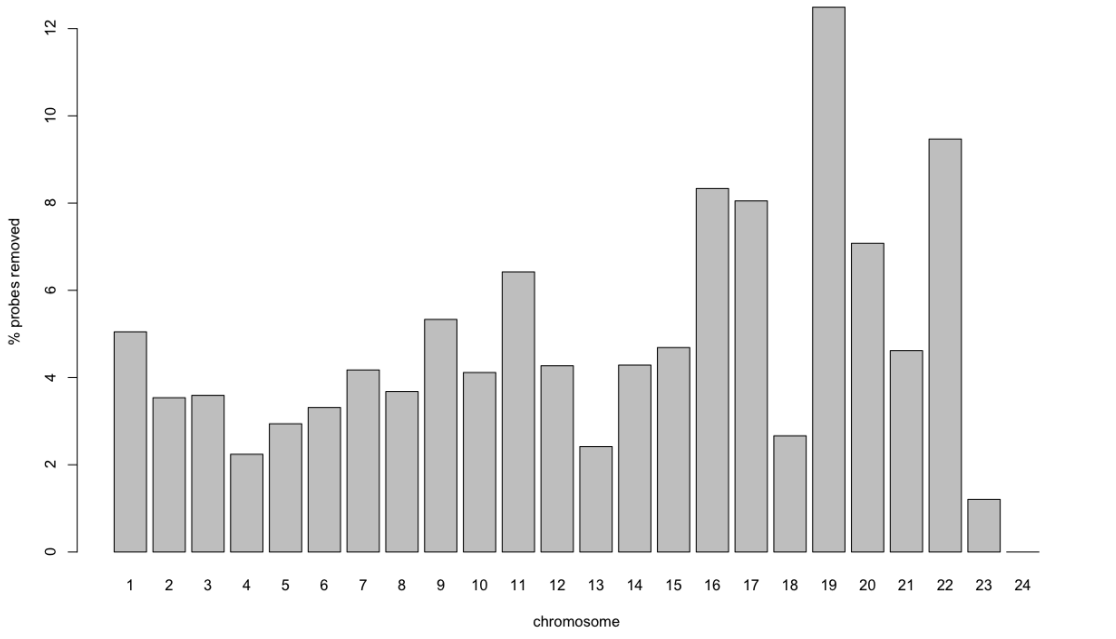


Figure 2-39 Proportion of probes per chromosome removed using Mann-Whitney p values

It was important to ensure that relatively even coverage across the genome was maintained after probe removal. Above (see **Figure 2-39**) the proportion of probes removed per chromosome across the array-CGH platform is shown. Overall less than 15% of the probes from any chromosome were removed and for most chromosomes less than 5% of probes were removed. Chromosomes 19 and 22 had the highest proportion of probes removed.

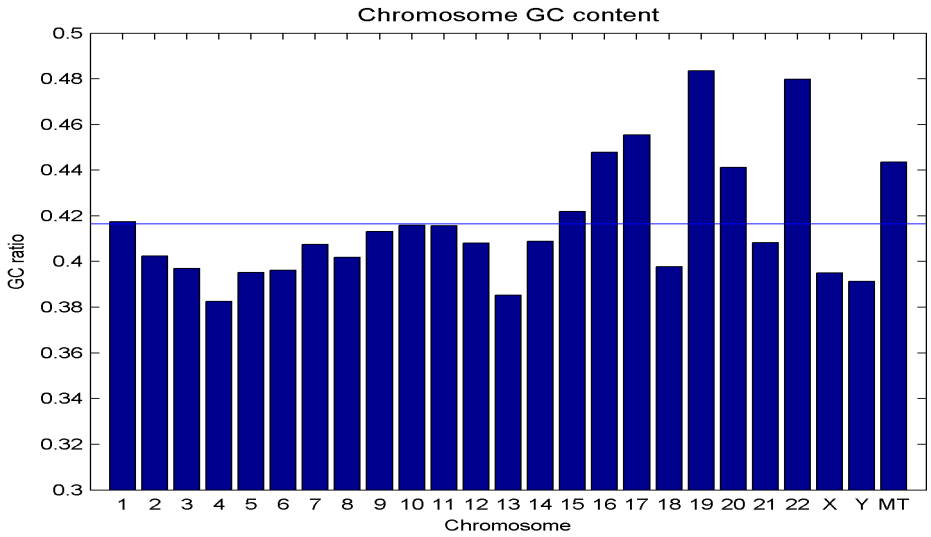


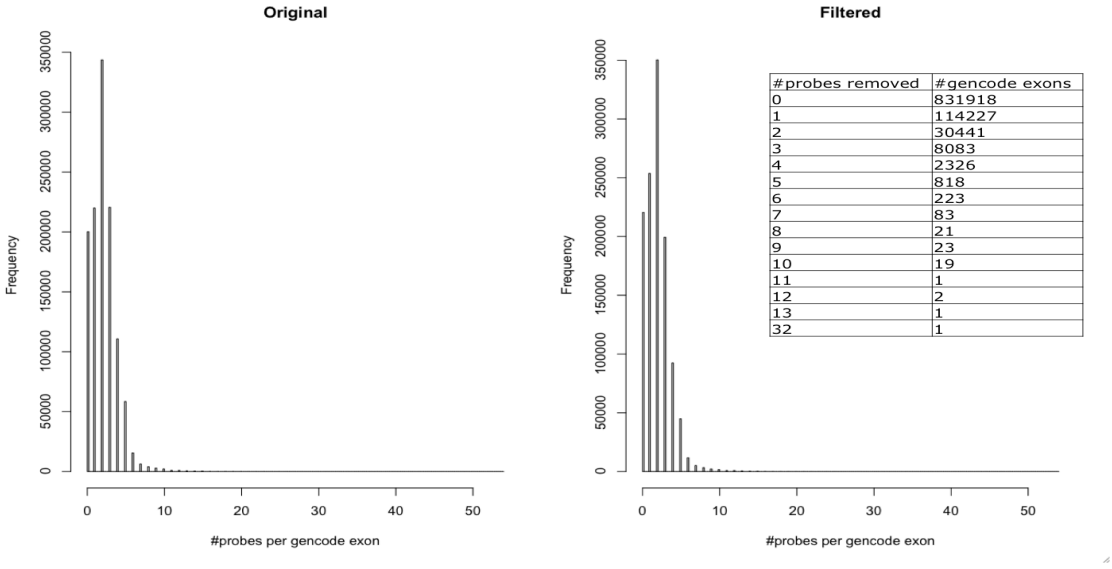
Figure 2-40 Average GC content per chromosome [blog.kokocinski.net].

Interestingly when comparing the proportion of probes removed per chromosome with the average GC content per chromosome a remarkable similarity was observed (see **Figures 2-39 & 2-40**). Overall the GC rich chromosomes tended to have more probes removed using the Mann-Whitney p values for probe performance between the two sample sets. This is not too surprising when you consider that PCR amplification of GC-rich DNA sequences is often more difficult due to the formation of secondary structure such as hairpins and a higher melting temperature [175]. Therefore where DNA quality and concentration vary the amplification of GC-rich DNA sequences is likely to be affected resulting in poor performance for probes with a high GC content.

Number of Genic Probes Removed

On top of ensuring overall even probe coverage across chromosome, perhaps more importantly it was necessary to ensure that the coverage of individual genes and exons would not be compromised by removing problematic probes. We compared the number of array-CGH probes overlapping every exon from the GENCODE gene set (version 17) before and after probe removal.

Table 2-11 The number of probes removed and the number of GENCODE exons with that number of probes removed after probe removal.



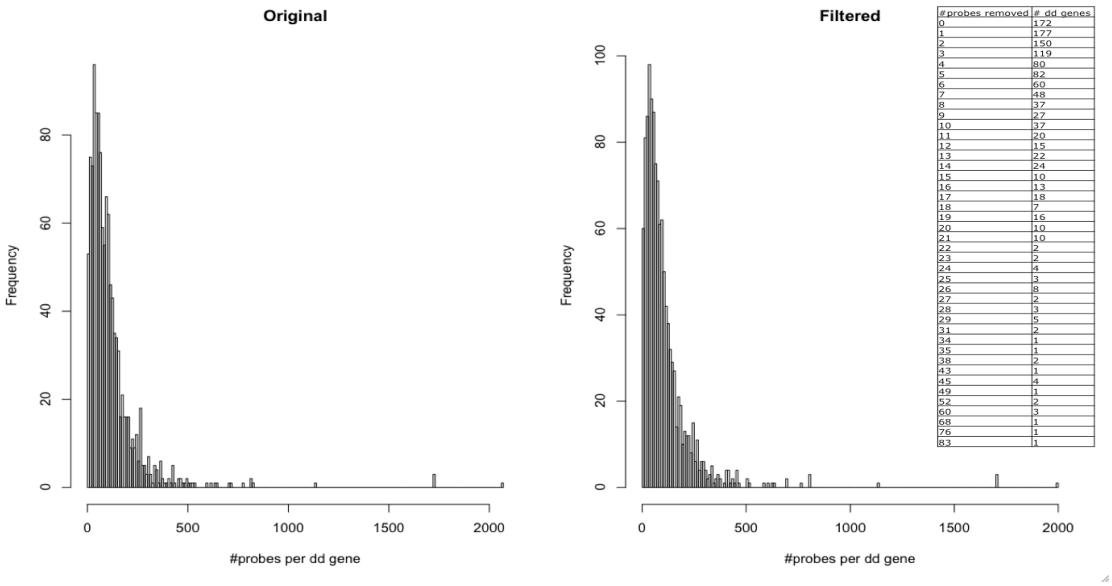
#probes removed	#gencode exons
0	831918
1	114227
2	30441
3	8083
4	2326
5	818
6	223
7	83
8	21
9	23
10	19
11	1
12	2
13	1
32	1

Figure 2-41 Number of array-CGH probes per GENCODE exon before (left) and after (right) probe removal.

The overall exon coverage of the array-CGH platform was not severely compromised by removing 4.7% of probes on the array due to their differential performance between sample sets of differing DNA quality (see **Figure 2-41** & **Table 2-11**) The majority (84%) of GENCODE exons had zero probes removed, 11% and 3% of targeted GENCODE exons had only a single probe or two probes removed respectively. Collectively less than 2% of all targeted exons had greater than two probes removed and the median number of probes per exon remained at 3 before and after probe removal (see **Figure 2-41**).

Next we compared the number of probes per DDG2P entry before and after probe removal to ensure that the diagnostic power of the array-CGH platform for genes previously associated with developmental disorders was not compromised by the probe removal strategy (see **Figure 2-42**).

Table 2-12 The number of probes removed and the number of DD gene with that number of probes removed after probe removal.



#probes removed	# dd genes
0	172
1	177
2	150
3	119
4	80
5	82
6	60
7	48
8	37
9	27
10	37
11	20
12	15
13	22
14	24
15	10
16	13
17	18
18	7
19	16
20	10
21	10
22	2
23	2
24	4
25	3
26	8
27	2
28	3
29	5
31	2
34	1
35	1
38	2
43	1
45	4
49	1
52	2
60	3
68	1
76	1
83	1

Figure 2-42 Number of array-CGH probes per DDG2P gene before (left) and after (right) probe removal.

Overall 14% of DDG2P entries had zero probes removed and 41% of had less than 3 probes removed. More than 80% had less than ten probes removed and 94% had less than twenty probes removed (see **Table 2-12**). The median number of probes per DD gene remained at 97 before and after probe removal.

Based on the results presented above it was decided to implement the probe removal strategy using the Mann-Whitney p values resulting in removing 4.7% of the probes on the array-CGH platform as a standard step in the array-CGH analytical pipeline for the DDD project. Because this change involved removing data points from the raw (input) data before any analytical processing could occur it was necessary to increment the main acgh-pipeline version number (see **Chapter 3**) and re-run the entire pipeline for all data processed to date.

Filtering Results Before and After Probe Removal

The main pipeline version number was incremented from 0.1 to 1.0.0 and the entire pipeline was re-run for all available data sets. Originally there were 1,395 QC passed array-CGH datasets using pipeline version 0.1, of which 99.5% passed sample QC using the new pipeline version (1.0.0). Seven samples failed at post calling QC as, due to the removal of probes, some CNV call characteristics (e.g. number of calls, breakpoint locations, mean log2 ratios, wscore) were understandably slightly different between the pipeline versions and due to the nature of the post calling QC methods (see **Methods**) slight differences in the characteristics of individual CNV calls can result in sample and data QC

differences in both directions (the QC cut-offs are not static, fixed parameters). However, overall for the vast majority of samples (1388/1395) the QC results for both sample and data quality checking remained the same after the pipeline version was incremented to 1.0.0.

Table 2-13 Summary statistics for the number of CNV calls made using pipeline version 0.1 and 1.0.0

Pipeline Version	# Calls	# Rare Calls	# Flagged	# Reported
0.1	330803	92532	411	57
1.0.0	301409	78965	225	39

Using pipeline version 1.0.0, which included the removal of poorly performing array-CGH probes prior to CNV detection, there were less CNV detections made overall with a median of 217 CNV calls per sample compared to 238 per sample when using pipeline version 0.1 (see **Figure 2-43**). The number of rare calls also decreased slightly when using pipeline 1.0.0, with a median number of 56 compared to 66 when using pipeline version 0.1. The number of CNVs flagged for clinical review was decreased almost two-fold when using the new pipeline version suggesting that recurrent CNV calls at problematic regions (e.g. FMR1, SHANK3 and CHD7) may have been avoided. However, a relatively large number of CNVs that had previously been reported during the manual step and had been deposited in DECIPHER using results from pipeline 0.1 were apparently no longer called using the pipeline version 1.0.0 (see **Table 2-13**).

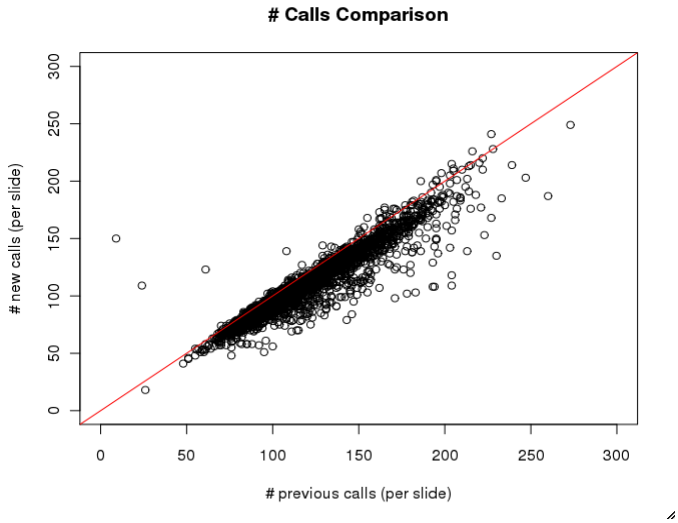


Figure 2-43 The number of CNV calls made per slide using pipeline version 0.1 (previous calls) vs. pipeline version 1.0.0 (new calls).

As expected each slide had slightly less CNV calls using the new compared to the old pipeline. For the new pipeline version results, 4.7% of the array-CGH probes

had been removed and not used for CNV detection therefore it would be more difficult or even impossible for the new pipeline to call CNVs including signal or dominated by probe signals that were no longer used for CNV detection.

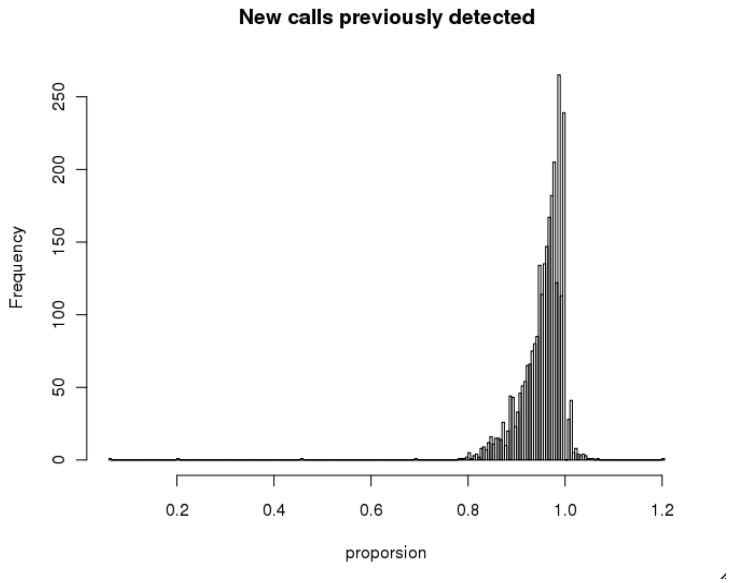


Figure 2-44 The proportion of new calls (pipeline 1.0.0) previously detected (pipeline 0.1) per sample.

Due to the removal of probes across the array-CGH platform the exact breakpoint (change point) location and change point interval (distance) are liable to some change between pipeline versions. Overall the majority of samples show very high similarity between pipeline versions, with the proportion of previously detected CNV calls tailing off very quickly below 90% (see **Figure 2-44**). The small number of samples with greater than 100% previously detected CNV calls can be explained by CNV intervals that were split into more than one event in the new pipeline compared to the old pipeline results.

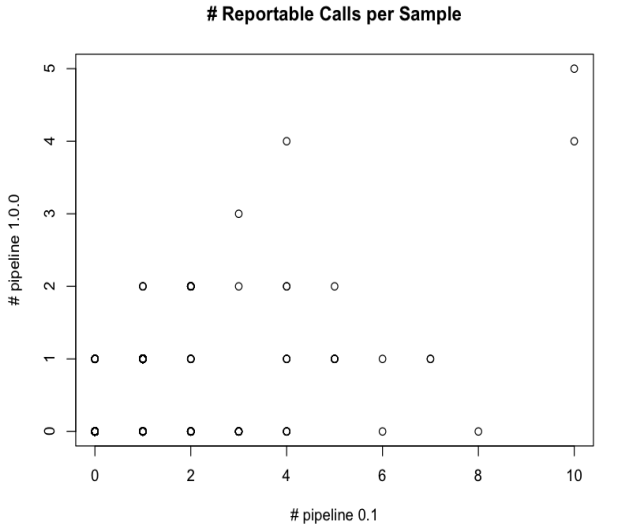


Figure 2-45 The number of CNVs flagged for clinical review per sample using pipeline version 0.1 vs. pipeline version 1.0.0.

Overall the number of CNVs flagged for clinical review was decreased almost two-fold following the pipeline version increment (see **Table 2-13**). Furthermore the number of flagged CNVs per sample was decreased between versions (see **Figure 2-45**), indicating that problematic CNV calls recurrently made in relatively large numbers of samples had been avoided in the new pipeline version results.

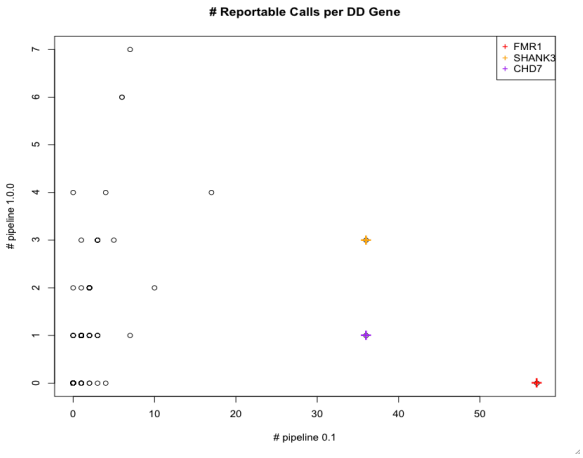


Figure 2-46 The number of CNVs flagged for clinical review per DD gene using pipeline version 0.1 vs. pipeline version 1.0.0.

The number of CNVs flagged for clinical review per DD gene was decreased considerably between pipeline versions. This was encouraging as the reason for apply the probe removal strategy and incrementing the main pipeline version was the observation of relatively large numbers of flagged CNVs in specific DD genes which, after manual review and although being high quality CNV calls,

were deemed to be spurious results and not associated with specific patient phenotypes.

The CNV calls in the three main genes that had initially indicated the problem (FMR1, SHANK3 and CHD7) has decreased dramatically between pipeline versions (see **Figure 2-46**). However, there were some additional genes that had not previously been observed as having high recurrent rates across samples that had also seen a decrease in the number of times they were flagged for review between pipeline versions. This suggested that the probe removal strategy had worked rather well and on top of removing problematic calls at FMR1, SHANK3 and CHD7 had also resulted in decreasing the rate of recurrently flagged CNVs in other DD genes. These recurrently flagged CNVs in additional DD genes were highly likely to also be spurious due to both the relatively high (implausible) rate across samples and the fact that the behaviour of the probes instrumental in detecting the previously flagged CNVs were significantly associated with that of the known problematic probes.

CNV Calls in CHD7, SHANK3 & FMR1 between pipeline versions

The most straightforward and logical way of assessing the effect of the probe removal strategy was to look at the number and type of CNV calls made at the three defined problem genes (FMR1, SHANK3 and CHD7). Reasoning that if the strategy had been successful in removing poorly performing probes that were the cause of false positive CNV calls in samples with no associated phenotypes these CNV calls would no longer be made.

Table 2-14 The number of CNVs flagged of clinical review at CHD7, SHANK3 and FMR1 between pipeline version 0.1 and 1.0.0.

Gene	**# Flagged pipeline 0.1**	**# Flagged pipeline 1.0.0**
CHD7	35	1
SHANK3	36	3
FMR1	57	0

Above (see **Table 2-14**) that the number of flagged CNVs at CHD7, SHANK3 and FMR1 show a massive decrease for the results from pipeline version 1.0.0 compared to 0.1. There were no CNV calls remaining at FMR1, 3 at SHANK3 and 1 at CHD7. It should be noted that not all probes in these problematic genes were defined as poor quality and we should expect to still be able to call good quality CNV calls in FMR1, SHANK3 and CHD7 using the remaining probes (74, 43 and 202 respectively).

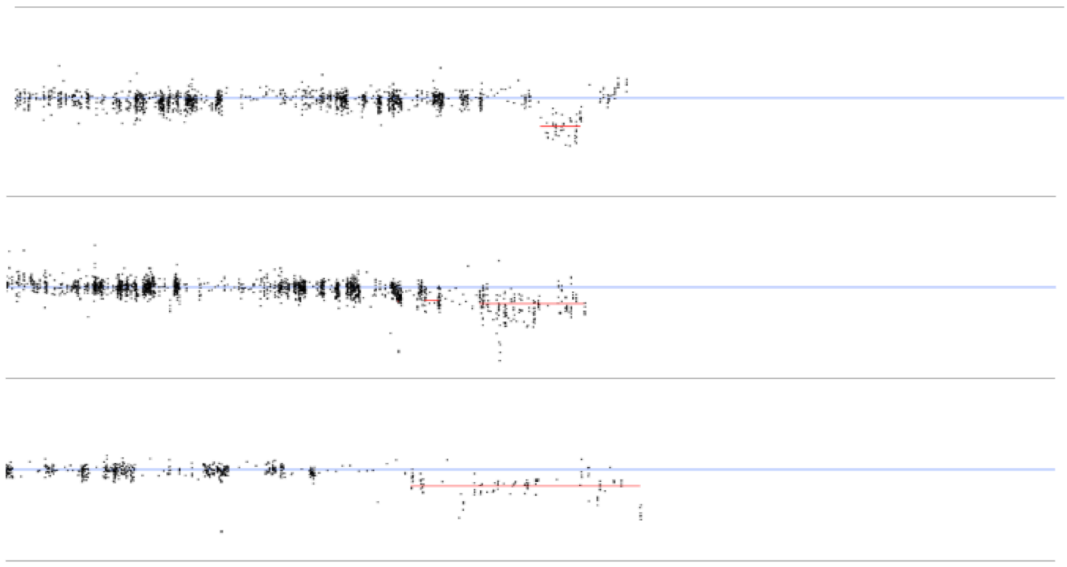


Figure 2-47 Remaining SHANK3 CNV calls in pipeline 1.0.0 results.

The above plot (see **Figure 2-47**) shows images of the log2 ratio profiles across the three remaining SHANK3 CNV calls, the red bar indicates the change point interval defined by CNsolidate. All problematic array-CGH probes have been removed and the probes indicating single copy deletions in the three individual samples are all deemed high quality. The data noise profile of the background and change point locations in all three samples indicate high quality CNV calls. Furthermore all patients presented with Autism-like phenotypes and deletions at SHANK3 are the most common cause of Phelan-McDermid syndrome, a syndrome that often includes a degree of autism [176].

Previously Reported Variants

Fifty-seven CNV calls had been reported via DECIPHER to the clinical diagnostic teams using results from pipeline 0.1. Of the 57 previously reported CNVs, 39 were called using pipeline 1.0.0 with exactly the same breakpoint locations, 17 were called with only slight differences in breakpoint locations between pipeline versions and one previously reported CNV call at CHD7 was no longer present in the 1.0.0 version results.

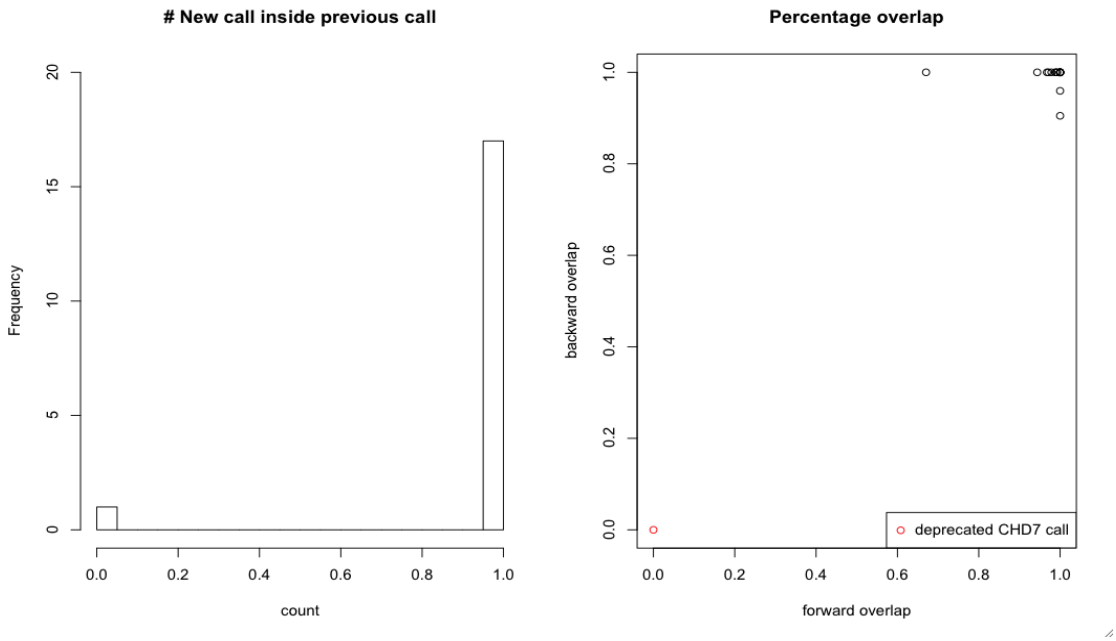


Figure 2-48 Left – the count of the number of CNV calls made by pipeline 1.0.0 inside a reported CNV called using pipeline 0.1; Right – the forward overlap the (old to new CNV call) vs. the backward overlap (new to old CNV call).

The majority of the CNV calls show high degrees of similarity in breakpoint locations between the two pipeline version numbers (see **Figure 2-48**). Two CNVs show slight decreased backwards overlap values, indicating that the new CNV calls are slightly larger than the old CNV calls. One CNV calls shows a marked decrease in the forward overlap value indicating the change point interval is smaller in the CNV call made using the new pipeline version. One CNV call shows zero forward and zero backward overlap values meaning that it missing from the new pipeline version results, however the CNV call was also deprecated in DECIPHER due to an alternative diagnosis and no CHARGE, KAL5 or IHH like phenotypes displayed in the patient which could result from a CNV at CHD7 [177].

2.4 DISCUSSION

Conclusions - CNsolidate

We have developed a change point detection package, CNsolidate that makes use of multiple change detection algorithms and an expert voting system with the goal of increasing detection rates and detection consistency across multiple data sets. The benefit of using multiple change point detection algorithms to increase ones confidence in individual detections using a naive voting system (consensus) and an expert voting (integrating prior knowledge) approach has been demonstrated. Evidence was presented, in the form of an analysis of receiver operator characteristics, to show that by using its combined approach, CNsolidate, delivers improved performance compared to any of its individual components algorithms.

Examples of using the score adjustment strategy to allow detection quality scores to be ranked between data sets based on a desired level and type of truth has been shown. Again using a ROC analysis we demonstrated the ability to set a single threshold on the detection quality measure (adjusted wscore) to obtain a consistent level of performance across multiple data sets (50 replicates of the DDD control sample NA12878). Moreover, by comparing the performance of the adjusted wscore for defining detection quality against a number of other approaches to detection filtering we found that the wscore generated by CNsolidate performed best for predicating CNV call quality. CNsolidate contains a number of individual change point detection algorithms, including previously published and novel methodologies, however it is made available as a stand-alone analytical package with no external dependencies. This not only provides a given user with a simple one-step installation procedure but also ensures that the consistency of the package is maintained. Changes or improvements, made externally, to any methods contained inside CNsolidate are likely to be only updated within CNsolidate once a review process, relating the cost-benefit and maintenance of analytical consistency, has been undertaken.

The large number of parameter definitions within CNsolidate are contained inside a single configuration, allowing modifications to be made with relative ease. CNsolidate contains a large number of data simulation methodologies, semi-automating the tuning procedure for individual algorithms and the associated weighting functions (see **Appendix**). Overall we found that **1)** combining information between multiple change point detection algorithms; and **2)** integrating prior knowledge when generating a confidence score (wscore); resulted in an increase in detection performance and consistency across relatively large numbers of datasets.

Conclusions – CNV-tagging SNPs

We have developed a novel analytical approach for the tracking of array-CGH data using copy number tagging SNPs. This was important to the DDD project to provide a level of confidence that the data going through the analytical pipelines were from the expected individual (sample). The method developed uses a probabilistic approach to determining sample mismatches. By applying this approach to control data it was possible to detect and resolve a known sample mismatch. Additionally, using random data sampling, the presented

discriminatory power of the approach was relatively good for high quality (low noise) data and as data quality decreased the discriminatory power of the tracking values also decreased. The method also constantly updates the parameters of the model using a fast lookup and recalculation of some necessary parameters. It is expected that as more data are added into the model the discriminatory power of the tracking values are likely to improve overall.

Because of the large data sets in question this functionality relies heavily on the ability to extract the necessary information in a timely fashion. We have developed a novel data storage and accessing method using a specific type of binary format and random access data reading methods. These accessor methods display impressive performance characteristics and allow the lookup and update to execute in a fraction of the time that would otherwise be required.

Conclusions – CNV Consensus

The CNV consensus reference set has been created using a number of high quality studies into common genomic variation. This set, being made up of predicted high quality data sets, can aid the accurate filtering of variants across the genomic range. By combining the data sets into the CNV consensus we achieve good coverage of common variation across a large genomic size range. Having included both sequencing and array-based studies into the CNV consensus the positional information from small genomic regions up to the larger structural rearrangements has been included. By combining these different data sets it was possible to not only define regions of common genomic variation but also give an estimate of the population frequency at each CNVE. This is important for accurate and flexible filtering procedures, for example having observed a variant only once (singleton) in a study containing 5,919 samples is very different to observing a variant multiple times across a number of different studies.

Providing an accurate estimate of population frequency for common CNVEs allows filtering methods to group variants into different categories using different frequency cut-offs. For example, one could define different frequency cut-offs to classify a test variant into common, rare or novel CNV types. We have provided a general approach for calculating the positional similarity of test CNVs to CNVEs contained inside the CNV consensus reference set, maximising both the forward and backward overlap values to find the frequency of the most similar CNVE in terms of their genomic coordinates. Additionally, the CNV consensus provides a state type value. This is important because, having observed only duplication type variation at a given genomic location in studies of common CNV does not necessarily mean that a deletion at the same location would not be harmful. There are a number of genes in which an increase in dosage (duplication) is tolerated whereas a loss of one (heterozygous deletion) or both (homozygous deletion) copies can result in extreme phenotypes. The ACP5 gene when homozygously deleted can cause certain pathological states such as Sjogrens syndrome [178], whereas certain pathological states can result from the duplication of other genes, for example the duplication syndrome of the MECP2 gene [179]. Importantly the approach developed for assigning population frequency estimates to test CNVs takes this information into account and only compares deletions to exclusively deletion or deletion and duplication sites; and

only compares duplications to exclusively duplication or deletion and duplication sites.

Finally, the CNV consensus set can be visualised using online resources such as DECIPHER, ENSEMBL and UCSC. Additionally, the CNV consensus set is provided as a data source, along with a number of functions relating to CNV interpretation, within the CNsolidate package. We explicitly provide two different approaches to variant filtering using either the separate merged study sets or the complete combined CNV consensus. For the DDD analytical pipeline we preferentially use approach one to variant filtering, comparing any test CNV location against all individual CNVE sets from the CNV consensus. Using this approach ensured that problems encountered when combining information across studies of different sizes and effective genomic resolution was avoided. For example, adding the number of samples together to generate a combined frequency measure can be problematic if the operational genomic resolution of each study is not well understood. For illustration purposes imagine the case where a single small deletion is detected in a high resolution study with 20 samples, if we use the "in study" frequency we calculate the value at (0.05). However if we combined this information with a larger study containing 5000 samples that did not have the capacity to detect CNVs at the same position and did not take that information into account we would incorrectly assume a "cross study" frequency of 0.0002. This is a general problem with combining information from large numbers of studies to use as reference sets for filtering common genomic variation. The CNV consensus aims to address some of these issues and provides two approaches to matching CNVs with population frequency information. Furthermore for approach 2 we have estimated the functional resolution of each platform used in the CNV consensus and, for each genomic location, only combine information across studies that could theoretically have detected variation at the particular locus. However, the operational detection resolution is hard to estimate as, for example, each study used different analytical methodology (change point detection algorithms) displaying differential performances.

CNV Filtering

We have designed a CNV filtering strategy for flagging variants of potential clinical significance and implemented a fully versioned and adaptable CNV filtering pipeline for all CNV calls made by CNsolidate for the DDD project. This pipeline is of critical importance to the ability to feed results from the DDD project back to the clinical teams across the UK and Ireland. We make use of a dedicated resource (DDG2P) linking types of variation to genes previously associated with genomic disorders and additionally providing gene to phenotype associations using the fixed term HPO ontology [152]. Key to the ability to link CNVs with specific genes contained inside the DDG2P is the mode and mechanism annotations, meaning that the effect that a CNV overlapping any DD gene might have can be predicted based on the copy number state of the CNV.

The filtering pipeline flags CNVs for clinical review using one of two different routes that we term the VOUS and the DD gene routes. CNVs flagged via the VOUS route are large enough and observed at low enough frequency in the general population to be thought of as having general clinical significance. On the other hand, CNVs flagged via the DD gene route are very specific and display the correct copy number state to have a high likelihood of causing the phenotypes

previously described for the specific gene. We did not apply any phenotype matching, using the HPO terms to match patient records in DECIPHER to gene entries in the DDG2P, rather simply considered all flagged CNVs to be of general clinical interest and relied on the expertise of clinical specialists to interpret the meaning and importance of such CNVs with respect to patient information. Rather than attempting to automatically interpret the clinical significance of CNVs detected, the DDD project prioritises and tracks CNVs, enabling clinical interpretation, patient feedback and ultimately a potential diagnosis for patients and their family. Furthermore all information for detected CNVs across the DDD project cohort is maintained in dedicated databases and file systems as well as detailed patients records in DECIPHER facilitating the search for novel variants to gene function discovery in a research setting [180].

The overall results from filtering CNVs for clinical significance using two different versions of the acgh-pipeline have been presented. To ensure patient confidentiality and avoid ethical issues results were presented using global summery statistics rather than providing specific findings in individual patients. No incidental findings were discussed and no patient identifiable information was given. A specific technical problem noticed due to a relatively large number of flagged CNVs in specific DD genes was described, a detailed assessment into the cause was presented, a solution was implemented and evidence that the sensitivity for detecting CNVs using the array-CGH platform was maintained at an acceptable level was provided. Overall the performance of the clinical filtering component in the acgh-pipeline was initially relatively low in terms of total numbers of flagged variants. This was a deliberate choice by the DDD project since the manual review and feedback of variants is a time consuming process and the available resources (primarily clinicians' time) needed to be carefully balanced. Furthermore, the parameters chosen for flagging variants were initially set to be high stringency in terms of detection confidence to ensure all flagged variants were high quality resulting in a low false positive rate. As the DDD project progresses these parameters and filtering rules are likely to be relaxed, potentially increasing the diagnosis rate from CNVs. The clinical filtering component has been developed to allow changes in filtering logic to be applied with relative ease using version control and configurable parameters. Furthermore, the data storage structures have been designed to accommodate changes to the pipeline versions and analytical results obtained from the acgh-pipeline.

3 | DDD Pipeline Development

3.1 OVERVIEW

The DDD array-CGH pipeline is built to run in batch on a load sharing facility (LSF) and stores results in both a dedicated file system and database. A dedicated POSTGRESQL database and file system to store and maintain all analytical information generated by the DDD array-CGH pipeline has been developed.

The file system structure is key to the accurate maintenance of analytical information, more so than the database. The file system is a tree-like structure, based on a running integer using a pad and split approach. This ensures that the file system is structured and that each directory only contains a maximum of 100 directories. For storing results at scale it is important to ensure that single directories do not contain too many items. As the number of items in a directory increases the performance of simple file system operations, such as listing the contents of a directory, becomes severely compromised and has the potential to add considerable overhead to input / output (I/O) speed. The second last directory in the full path for each sample is the 'sanger id', followed by the specific experiment (array-CGH slide) ids; this allows all data for any given sample to be easily found. Additionally, the information contained under the directories for a given sample contains all the information stored in the database. This means that the database can be rebuilt from the file system at any given time. Furthermore the entire file system is backed up at regular intervals adding extra security against potential data loss.

There is a dedicated database and file system accessor and storage package, written purely in R, allowing querying of data across the file system and database tables using specific R class definitions. For the general user this hides a lot of the complexity within the data storage structure and allows, for example, simple queries to retrieve all analytical information for a sample using its sanger id as input to a single command inside the R interface. This is of particular use for others within the DDD project wanting to query the data but who have limited technical knowledge of the SQL language and the file system structure.

The structure of the SQL database is overall very simple and effectively only has eight independent tables with no explicit foreign key relationships. The decision to implement an unnormalisaed database structure was taken early on as the expectation was that the results would eventually be subsumed by a larger more complicated variant database housed at the Wellcome Trust Sanger Institute. Results from the DDD array-CGH pipeline are now being incorporated into a new database structure that is planned to drive further analytical work for the DDD project. CNsolidate is an integral part of the DDD array-CGH pipeline, containing the greatest number and most complex analytical software components of all the internal DDD analytical pipelines.

3.2 IMPLEMENTATION

The array-CGH pipeline implemented for the DDD project is an automated analytical pipeline including the following main analytical steps:

- Archival
- Normalisation
- Detection
- Annotation
- Quality Control
- Filtering
- Storage

All steps are executed in batch using a single command and all results are archived in a reliable and ordered way. At the end of each batch run an automated summary report is generated detailing technical information relating to time taken and any potential errors encountered during data processing.

The entire pipeline is very large and complex but can, for the most part, be used as a "blackbox" unless something specific goes wrong. The execution is driven via a single input file ("keyfile") and each individual step can be executed using this keyfile as the input. The keyfile is generated by querying the DDD laboratory management system (LIMS) the first time an acgh plate is run, however it can also be "re-generated" from the analysis database in some cases. For example for the automated re-run of any processing failures using the "checker" script a new keyfile is generated containing the required information and used as the input to the main pipeline execution scripts. This "checker" script is fully automated and runs weekly under the time-based job scheduler CRON.

Version Control

A very important aspect of developing maintainable software is that of version control, especially if multiple developers are working on the same code base. Although I was the main developer of the acgh-pipeline for the DDD project others have contributed to specific areas (methods not described in this thesis) and it was important that changes made to the main code base by any one individual did not affect the work of others. Therefore it was necessary to ensure a reliable way of maintaining software components and resolving any conflicts in code changes before commits were made. To do this we choose to use the version control system Git and to create a dedicated software repository named "acgh-pipeline".

There is a relatively big following and preference for using Git compared to other version control systems at the WTSI and in the Bioinformatics community generally. The main advantage of Git compared to other version control systems such as Subversion is its relative simplicity of use, for example its branching and merging functionality are very intuitive and easy to perform.

Repository Layout

One of the first tasks to perform and decisions to make when creating a software repository is that of the package layout. It is important to establish some general principles and consistent approaches to where in the code base different software components are placed. If the project is confined to a single language and has a clear overall goal this can be very straight forward. For example the development of a web tool using a language such as Java might simply use the project layout dictated by the software environment and framework chosen (e.g. a Spring Source project developed in the Eclipse IDE).

On the other hand when projects include a number of different programing languages and different development strategies for software components this can be quite challenging and has the potential to cause considerable confusion. Generally the best approach is to try and keep things as simple and clear as possible, meaning that ideally any developer who obtains a copy of ("clones") the repository can quickly and easily understand the project layout.

Fortunately there are some standard ways of laying out software packages and it is normally relatively straight forward for an experienced software developer to understand any project layout. As the acgh-pipeline is made up of a large number of different software components written in a variety of different programming languages it follows a standard approach and the code base is separated in a clear and understandable way containing the following items at the project root:

Makefile
README
-- init
-- src
-- util

Firstly the root contains two files that are often found in software package distributions, a "Makefile" that controls the installation of all software components, and a "README" file that lists important information and provides guidelines on installation and usage. Next we find three directories with standard names and purposes. The "init" directory is a generic place where useful software code can be placed, it is normally where useful scripts rather than self contained software components will reside. Under the "init" directory there are normally several subdirectories named as the programming language they contain (e.g. perl) and within these subdirectories the layout is for the most part free to choice, although of course should still ideally be logical. Next the "src" directory is the place where complied project elements are normally placed and should contain complete self-contained software components, for example libraries written in C, C++ or R should be included inside this directory. Finally the "util" directory is where utility items should be placed and is normally used for configuration and execution pipeline elements.

Software Installation

The installation of the entire software package for the acgh-pipline has been made relatively easy by including a dedicated "Makefile" within the package distribution. The Makefile is a script used by the functionally of the unix command "make". All installation locations and resource requirements are contained within the Makefile as variables and all installation steps are contained as rules to the "make" command.

To install the acgh-pipeline all that is required is that the user types "make" on the command line within the acgh-pipeline package root. However, it is certainly required that the information within the Makefile is set correctly. This can be achieved by either editing the Makefile in place or by passing the variables specific values to "make" on the command line. So long as all the installation paths, required software versions and pipeline resources are available, "make" will install all software package components into the correct locations under the desired installation root and will only return successful if no step in the installation procedure failed.

There are 6 rules contained inside the Makefile that control the entire installation procedure. These rules are named:

- createcnsoldiate
- createpipeline
- copyscripts
- createconfig
- installRpackages
- clean

First, the createcnsoldiate step attempts to make the installation root, R library root, obtains and unzips a fresh copy of the acgh-pipeline code base via the Git command archive. Next the createpipeline step sets up the software package layout by attempting to make all subdirectories beneath the installation root. Then the copyscripts step copies all pipeline scripts into place and uses the "sed" command to modify items in the scripts relating to the location of the installation root. Next the installRpackages step installs all included R packages (internal and external) into the defined R library location. Finally the clean step removes all "step execution files" generated by steps 1 to 6. Each step creates a "step execution file" using the UNIX "touch" command, indicating that the step has been a success. Each individual step depends on the existence of the "step execution file" from the previous step and will not execute if that file is not present. The last step (clean) removes all "step execution files" including its own.

The precise version of the acgh-pipeline installed is controlled using the Git "tags" functionality. A Git tag is used to label a specific point in the version control history and that point becomes a stable pipeline version. This tag is set as a parameter in the Makefile and the createcnsoldiate step uses that tag to obtain the correct version of the acgh-pipeline. This means that stable pipeline versions can be maintained while at the same time working on future releases using the same Git repository.

Pipeline Configuration

All pipeline parameters are configurable using a single JSON formatted config file that is intrinsically linked to a pipeline version (/acgh-pipeline-VERSION/templates/config.json).

Below are the available parameters and their descriptions:

- **version**: *pipeline version*
- **git_version**: *git tag of pipeline version*
- **project**: *project used for farm bsubs (defaults to ddd)*
- **genome_assembly**: *current genome build*
- **R.version.main**: *version of R to use (defaults to /software/R-2.13.0/bin/R)*
- **paths**:
    - **load_all_packages**: *the command to load all ddd R packages*
    - **ddd_data_dir**: *the output root for the file system directory structure*
    - **ddd_gene_list**: *the location of the current dd gene list*
    - **cnv.execute.all**: *location of main pipeline driver script*
    - **cnv.start**: *location of analytical flow (order) control script*
    - **cnv.execute**: *location of normalisation, detection, annotation & tracking script*
    - **cnv.detection**: *location of main CNV analysis script*
    - **cnv.execute.seq.sample.check**: *location of sequenom family check script*
    - **cnv.track**: *location of CNV tracking analysis control scipt*
    - **cnv.insert.qc**: *location of QC control script*
    - **cnv.insert.variants**: *location of variant insertion control script*
    - **cnv.inherit.all**: *location of inheritance classification control script*
    - **cnv.filter.all**: *location of clinical reporting filter control script*
    - **vis.data.location**: *location of data browser output files*
    - **vis.acgh.script**: *location of data browser file generation aCGH data script*
    - **vis.snp.script**: *location of data browser file generation SNP data script*
    - **rbin_index1**: *location of binary index file for DDD aCGH array1*
    - **rbin_index2**: *location of binary index file for DDD aCGH array2*
    - **mad_bin_file**: *location of large index file for DDD control data*
    - **black.list**: *location of the list of probes to remove prior to CNV detection*
    - **snp.map**: *location of index file for SNP data*
    - **snp.family.link**: *location of index file for family relationships of SNP data*
    - **snp.directory.location**: *location of index file for SNP directory structure*
    - **seq_package_root**: *location of sequenom family check package*
- **headers**:
    - **key.file**: *keyfile header*
- **db**:
    - **analysis**: *name of analysis database*
    - **lims**: *name of LIMS database*
    - **decipher**: *name of DECIPHER database*
- **penncnv**:
    - **exescript**: *location of penncnv execution script*
    - **hmmfile**: *location of penncnv hmm configuration file.*
    - **pfbfile**: *location of penncnv pfb file*
- **qc.thresholds.acgh**:
    - **processed_dLRs**: *dLRs QC cut-off*
    - **wave_estimate**: *wave estimate QC cut-off*

    - **sig_inten**: *signal intensity cut-off*
    - **sig_noi**: *signal to noise QC cut-off*
    - **bg_noise**: *background noise QC cut-off*
    - **repro**: *reproducibility QC cut-off*
    - **posterior**: *tracking posterior QC cut-off*
    - **low_sens**: *low sensitivity QC cut-off*
    - **del_dup_cut**: *deletion/duplication ratio QC cut-off*
- **qc.thresholds.acgh.call**:
    - **adjw_score**: *adjusted wscore call QC cut-off*
    - **p_value**: *p-value call QC cut-off*
    - **del_cut**: *mean ratio deletion call QC cut-off*
    - **dup_cut**: *mean ratio duplication call QC cut-off*
- **qc.thresholds.acgh.call.filtering**:
    - **common_forward**: *CNV consensus forward overlap filter QC cut-off*
    - **common_backward**: *CNV consensus backward overlap filter QC cut-off*
    - **del_size**: *deletion size filter QC cut-off*
    - **dup_size**: *duplication size filter QC cut-off*
    - **non_inherit_size**: *size filter for non-classified CNV filter QC cut-off*

This configuration json object is passed around all pipeline control processes such that the information contained within the pipeline configuration is available to all analytical pipeline elements. This has a very useful consequence and means that all configurable pipeline parameter definitions only need to be set in one place and there is only one file that needs to be modified if parameters are to be changed. Overall there are a great many more parameters used by CNsolidate and the acgh-pipeline than appear in the config file, however all external resources and important pipeline parameters are contained in the json object and are configurable. Furthermore, additional config parameters can easily be added to the json object if they are required by any particular pipeline component.

Automatic configuration during pipeline installation

As described above the installation of the entire acgh-pipeline is controlled by a single file, the “Makefile”. This file contains all the information needed to install the package under any LINUX based operating system. Additionally it contains all the information needed to set up the pipeline for standard use, specifically it contains all external and internal resource requirements, software versions and the root of the result storage file system. When the pipeline is installed using the Makefile a configuration object is created and place into the correct location in the pipeline software installation path based on the information contained inside the Makefile. Although it is possible to edit this configuration after the installation procedure, it is not necessary and the specific pipeline resources requirements and installation locations only need to be defined once within the Makefile before installation.

3.3 EXECUTION

Below are the commands and an explanation of each step for running the acgh-pipeline. The pipeline is executed using a single "bsub" command that will run the pipeline from start to finish, however it is possible to run each individual step in the pipeline. Below are details of each of the main individual steps:

Main steps in acgh-pipeline:

- Generating "keyfile" containing links between acghplate, sanger_id and slide_id for data to be processed (normally an acgh plate).
- Archiving of image and feature extraction files for all data sets to be processed.
- Copy input files onto data processing location (/lustre).
- Execute acgh-pipeline:
    - Insert sample level information to database
    - Run Sequenom family relationship test and store result in database.
    - Run normalisation, CNV detection and annotation (CNsolidate) for slide1 & slide2 for each data set (sanger_id).
    - Run data tracking method (CNV tagging SNPs) for both files (using coordinates in db the sequenom data and the binary formatted data file output from CNsolidate).
    - Run inheritance classification (VICAR) for all detected CNVs in proband (currently using SNP genotyping data - moving to exome data in future).
    - Run CNV clinical reporting filter.
    - Run database insert for all variants (fully annotated including clinical filter results).
    - Run database insert for QC steps and apply QC criteria (pass and fail samples based on quality and data tracking values).
    - Run browser file generation (cBrowse input files).
    - Run processing checker script (make sure that any processing failures are detected and re-run).

Main execution command:

The acghpipeline software is versioned under GIT and installed in the /software/ddd/acgh-pipeline directory.

- All following examples assume that the acgh-pipline version is 1.0.0

The normal output location for the acgh-pipeline is /nfs/ddd1 that is only writable by user ddd.

Switch to ddd user and login to farm

```
ssh ddd-vm1

sudo -u ddd bash

ssh -i /nfs/users/nfs_d/ddd/.ssh/ddd_dsa farm2-login

bash
```

The entire pipeline can be executed using a utility, this utility is aliased in the ddd users bash profile and is executed using acghVERSION.

So for VERSION 1.0.0 the base command would be acgh1.0.0.

- It queries the LIMS using the acghplate_id and generates a keyfile that acts as input for all the subsequent steps.
- The keyword "execute-all" executes all the steps listed above from start to finish.

Main execution of acgh-pipeline

```
acgh1.0.0 execute-all --plateid 239361 --scannerroot
/nfs/ddd0/DDD_ScannerData/ --fedest /nfs/ddd1/ddd_acgh_fe_files/ --
imagedest /nfs/ddd1/ddd_acgh_images/ --processingdirectory
/lustre/scratch113/projects/ddd/users/ddd/DDD_ANALYSIS/ --outputdir
/nfs/ddd1/ddd_data/
```

This results in all data sets being processed, all output files being stored in the directory structure and all values being inserted into the database for all data sets on the acgh plate.

Running Individual Steps:

The main steps in the acgh-pipeline can also be run individually using the same utility, each individual step is executed as follows:

Generate a keyfile:

The main input to all individual steps is a "keyfile" as mentioned previously. The first step to run the acgh-pipeline is normally the generation of a keyfile based on an acghplate id.

Generate a keyfile for acghplate 239361

```
acgh1.0.0 keyfile --plateid 239361 > 239361_keyfile_`date '+%Y-%m-
%d'`.txt
```

The key word "keyfile" and the argument --plateid are required and result in a query to the LIMS to collect the relevant information in the appropriate format. The output is given to stdout and should be saved to a temporary file for use in the subsequent steps. In this case we simply name it as plate_id underscore "keyfile" underscore "now time" in year-month-day format using the UNIX date command .

Archive image and feature extraction files from ddd0 to ddd1:

The normal flow of data for the acgh-pipeline goes from directories under "/nfs/ddd0/DDD_ScannerData/" to "/nfs/ddd1/". The next step in the pipeline is to find the relevant files on ddd0, copy them to ddd1, carrying out an md5 checksum and delete the files from ddd0 if the checksums match. This is done for image files and feature extraction files separately. Again the main input is a keyfile and all data files associated which the slide ids contained inside the keyfile will be archived.

Archiving image files for a keyfile

```
acgh1.0.0 archive --keyfile 239361_keyfile_DATE.txt --scannerroot
/nfs/ddd0/DDD_ScannerData/ --imagedest /nfs/ddd1/ddd_acgh_images/
--type image
```

The key word "archive" and the arguments --keyfile, --root, --dest and --type are required. The --type argument with a value of 0 results in the archiving of image files only. By default the --root and --dest locations are checked to be exactly the same as above and if either is different this step will not execute. This is aimed to avoid any mishaps but this behaviour can of course be overridden if required.

Archiving feature extraction files for a keyfile

```
acgh1.0.0 archive --keyfile 239361_keyfile_DATE.txt --scannerroot
/nfs/ddd0/DDD_ScannerData/ --fedest /nfs/ddd1/ddd_acgh_fe_files/ -
-type fe
```

Changing the --type argument to 1 results in the archiving of feature extraction files only. Notice that the --dest location is different than for the image files.

Overriding the default archive behaviour

```
acgh1.0.0 archive --keyfile 239361_keyfile_DATE.txt --scannerroot
SOME_ROOT --dest SOME_DESTINATION --type image --override 1
```

Providing the --override argument with a value of 1 results in the checking constraints on the --root and --dest arguments being removed.

Copying data input files to lustre:

The archiving of image and feature extraction files results in all data files for an acghplate being stored in the destination directory under a sub directory named by the acghplate id. Each data file is named by the slide id it is associated with. Any keyfile contains the number of rows, each of which contains the acghplate id and the slide ids for a given sample (sanger_id). This step copies all the relevant files from the storage directory to lustre ready for data processing.

Copying data files to lustre

```
acgh1.0.0 copy-to-lustre --keyfile 239361_keyfile_DATE.txt --
storageroot /nfs/ddd1/ddd_acgh_fe_files/ --processingdirectory
/lustre/scratch113/projects/ddd/users/ddd/DDD_ANALYSIS/
```

Again there are some checks in place and this stage to avoid any would be mishaps. It is ensured that the --storageroot argument contains "ddd1" and that the --processingdirectory contains. This behaviour can be overridden using the --override argument with a value of 1.

Executing the main CNV analysis step:

This is the main analytical step in the pipeline and executes all the heavy data processing (it will take approximately 2.5 hours to complete). The step includes, insertion of sample level information into the database (sample_info table), data normalisation, cnv detection, cnv annotation and data tracking.

Executing the main CNV analysis step

```
acgh1.0.0 execute-cnv --keyfile 239361_keyfile_DATE.txt --inputdir
/lustre/scratch113/projects/ddd/users/ddd/DDD_ANALYSIS/239361/ --
outputdir /nfs/ddd1/ddd_data/
```

The keyword "execute-cnv" and the arguments --keyfile, --inputdir and --outputdir are required. All files related to the slide ids inside the --keyfile are expected to be inside the --inputdir. A job array is submitted where each element is a row in the --keyfile (limited to 300 activate jobs). Each element of the job array submits a sequence of bsubs and all data processing occurs in "/tmp" space. After completion all output from this step is moved to its dedicated storage location under the main --outputdir (normally /nfs/ddd1/ddd_data/).
The --outputdir argument specified at this step is checked against the value inside the pipeline configuration file and if they do not match the step will not

execute. This behaviour can be overridden using the --override argument with a value of 1.

Executing the variant insertion and clinical reporting filter:

The next step in the pipeline is to insert all the "call QC" passed variants into the database (production analysis_live) and run the clinical reporting filter to flag variants for clinical reporting.

```
acgh1.0.0 execute-filter --keyfile
239361_keyfile_DATE.txt
```

This step executes a job array where each element is a row in the keyfile. All sample level information (most importantly file locations) is already inserted in the database. First each job looks up the location of the variant output files on disk from the database and attempts to insert all variants passing the QC criteria (found in the config file). The insertion includes inserting elements into the variant, acgh_variant, acgh_run and acgh_run_acgh_variant tables. Next all of the inserted variants (acgh_variant) are passed though the clinical reporting filter, this results in an update of the "filter_flag" value in the "acgh_variant" table.

Executing the sample level QC step:

The next step is the sample level QC for all data sets inside the keyfile.

```
acgh1.0.0 execute-qc --keyfile 239361_keyfile_DATE.txt
```

This step applies the defined QC criteria (see config file) for slides first, if either slide fails QC both slides (the sample) fails. Additionally, where there are replicates of the same sample (sanger_id) only the best quality data set (based on specific QC parameters) is passed. Furthermore, where we have a replicate person with an alternative id (alternative sample) only the 'best' quality sample for that person is passed.

Note: due to the way the QC steps work, if we re-run the QC across all datasets (but the number of datasets is different to previously e.g. additional data) the QC results for samples will be slightly different (i.e. small number of samples that previously failed may now pass and vis versa). This complication is due to the fact that the entire dataset as a whole is used during the QC steps. Due to this when we are applying qc to new data the entire dataset is interrogated but the qc result is only calculated for the new data. Therefore anytime we want to re-run QC for the entire dataset one need to be aware that the result may differ slightly to the previous QC classifications.

Executing the browser (cBrowse) file generation:

This is the final data processing step in the acgh-pipeline, it creates json formatted input files for the data browser. These include acgh, snp genotyping and exome data.

```
acgh1.0.0 execute-browser --keyfile
239361_keyfile_DATE.txt
```

Note: currently the dedicated output location for cBrowse file is "/nfs/ddd0/Data/cBrowse" however this directory currently has >2000 directories inside. This is an issue that needs to be resolved, the way the browser (cBrowse) searches for input files needs to be modified and then an appropriate directory structure should be created.

3.4 PERFORMANCE

Pipeline flow description

For each sample data set there are two experiment datasets (acgh slides) to be processed by the acgh-pipeline, each for which contains approximately one million data points. The overall pipeline flow is controlled using a "job array" indexed per sample, meaning that for each pipeline element (sample run) there are two independent processes executed (one for slide1 and one for slide2). The main analytical process for each independent slide consists of 26 individual steps. Each element in the job array relates to a sample (two slides) and the entire job array is limited to 300 simultaneously running jobs. The first general process in the pipeline is to collect all the information for each sample (two slides), input this information into the database and execute two processes (one for each slide) that perform the main analytical steps (26 steps). Additionally a third process is executed whose role is to wait until both main analytical processes have completed before allowing the pipeline to continue. We do not go into the full details of the pipeline flow control here but briefly it makes use of the LSF functionality of pre- and post- execution commands as well as the –w arguments to control the various job dependency states. Furthermore the memory usage is set using the –R arguments and request slightly higher memory than the predicated requirements such that processing failures are keep to a minimum. After both main analytical processes have run it is the third jobs role to execute the sample tracking, data quality control and clinical filtering steps for each person. The data for both acgh slides are required for the final steps to run correctly, which is why a relatively complex job dependency monitoring process was necessary to implement.

Main analytical process performance

The main execution steps of the acgh-pipeline consist of 26 individual steps. During processing of each slide a log file is written to disk containing information about which steps have completed and how long they took from start to finish. This log file is useful for monitoring results and pipeline performance, allowing any processing failures to be investigated in further detail. The log file will indicate for any processing failure the precise point in the analytical pipeline that the particular job (process) failed (stopped). There are a number of reasons why jobs can fail both due or not due to the acgh-pipeline software components. In the case where there was a real problem with one of the acgh-pipeline software components the log file provides a useful pointer for a developer to interrogate the relevant point in the pipeline in search for the precise code block causing the problem. A number of issues outside the control of the acgh-pipeline can cause processing failures, such as network outage, LSF issues and disk space limits.

This section will describe the performance of the acgh-pipeline in terms of time taken for each of the 26 main analytical steps across 8,852 datasets, each dataset contains approximately one million data points (acgh slide).

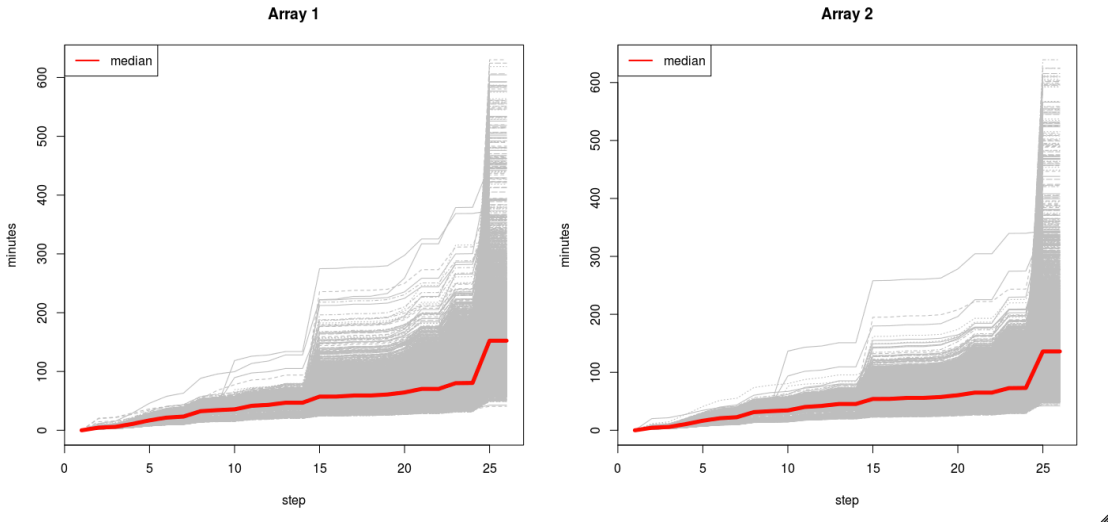


Figure 3-1 The time in minutes for the main analytical steps of the acgh-pipeline to run. Left – DDD acgh array1, Right – DDD acgh array2. Red – median time for each step across 4,426 slides of array1 and array2

The above plots (see **Figure 3-1**) show the cumulative time taken for each of the 26 steps contained in the main analytical process for array1 and array2 within the DDD acgh-pipeline. Overall the median time taken from start to finish of the main analytical steps are 2.5 and 2.3 hours for array1 and array2 respectively.

Probes (data points) across the entire genome are shared between the two DDD acgh arrays and both array1 and array2 contain 12 different chromosomes (making up the total of 24 chromosomes). One reason that array1 tends to take slightly longer to complete than array2 is the fact that array1 contains the sex chromosomes (chromosome X and chromosome Y). CNV detection on the sex chromosomes is often more difficult due to difference in the dose response (mean log2 ratio) between males and females, the normal copy number on chromosome X and Y in males is one whereas in females the normal copy number is two for chromosome X and zero for chromosome Y [181]. Furthermore the pseudoautosomal regions (PAR1 and PAR2) can cause additional problems to CNV calling. CNsolidate uses the approach of median normalizing each chromosome such that the normal copy number is represented by a log2 ratio centred on zero and and for chromosome X CNV calls made in the pseudoautosomal regions in females are excluded. Each sample on the DDD acgh arrays is run against a pooled DNA reference made up of 500 male individuals therefore the PAR1 and PAR2 regions normally falsely appear as deletions in females due to the median normalization of the chromosome X. This additional CNV detection complexity adds a small amount of time to the overall processing time of array1 compared to array2.

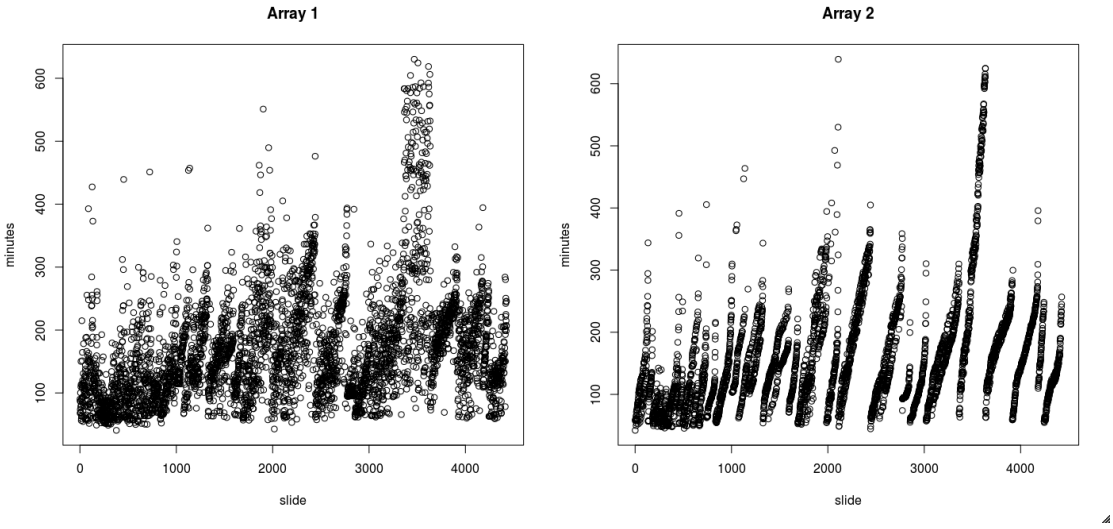


Figure 3-2 The total time in minutes for each slide to be processed through all main analytical steps ordered by date and time.

Above (see **Figure 3-2**) are two plots showing the total time in minutes taken from start to finish of steps 1-26 from 4,426 slides on array1 (left) and array2 (right) ordered by chronological time.

The noticeable periodicity is due to differences in compute resource usage institute wide at different points in time. The main limitation on processing time for the acgh-pipeline is the speed of reading and writing to and from the file system. The parallel distributed file system (Lustre) used at the Sanger institute provides high-performance I/O throughput however it is a shared resource. Due to particular pressures on project time lines or specific needs to perform large compute volume, the number of processes (jobs) hitting the file system can vary considerably at any given time. Noticeable trends in compute time for individual pipelines can often be correlated with large Sanger project data release dates. At other times it can just be a busy time with many people needing to process large data volumes or sometimes it can be due to a mistake made on the individual or project level where large numbers of erroneous jobs are causing problems institute wide.

The greater variance in compute time given the year-month-day observed for array1 compared to array2 can be explained by the fact that array1 contains both sex chromosome (X and Y) whereas array2 has no sex chromosomes. The added complexity mentioned previously in relation to calling CNVs on the sex chromosomes in females results in higher variance in compute time in array1 compared to array2.

3.5 DISCUSSION

A large-scale analytical pipeline incorporating the analytical methods described in this thesis has been implemented. This pipeline is fully automated, under tight version control and is in standard use in the DDD project for detecting and interpreting CNVs using a high-resolution array-CGH platform. There have been four stable acgh-pipeline version releases for the DDD project to date where specific changes (improvements) to analytical components have been made. One example of a change to the main pipeline version number is described in **Chapter 2** under CNV filtering where a major change to the CNV detection performance by excluding poorly performing probes prior to running CNsolidate was made. Other reasons for incrementing the main pipeline version number included, adding additional rules into data and call quality control, changes to the filtering logic for flagging CNVs for clinical review and an institute wide change to the version of LSF used at the WTSI.

The DDD acgh-pipeline was developed to be high throughput and achieve reasonable processing speed with a median of 2.5 and 2.3 hours for the DDD array-CGH array1 and array2 respectively. The pipeline is setup to allow 300 simultaneously running jobs by default. Although the total number of simultaneous jobs can easily be modified using a limit of 300 is predicated to be sufficient and deliver a reasonable time period to run all datasets for the DDD. Currently 5,651 samples (11,302 1 million probe slides) have been run through the latest version of the acgh-pipeline for the DDD project. All results have been generated and stored in both the file system and database structures and are actively being used to both provide clinical diagnosis to patient samples within the DDD and to further our understanding of the role that CNVs play in a number of rare disorders. Even if it were necessary to re-run the acgh-pipeline from start to finish for all DDD datasets (expected to be approximately 10,000 samples) it would take approximately 3.5 days to obtain all results and fully populate the data storage areas based on the median time taken to run the slowest step (the 26 main analytical processes on DDD array1).

I have provided a fully version controlled repository for all analytical code and software packages that make up the DDD acgh-pipeline along with detailed documentation explaining how to install and execute the pipeline in an automated fashion. By using the tagging (--tags) functionality provided by Git it was possible to provide specific frozen versions of the acgh-pipeline that can be easily cloned using the main pipeline version number. The acgh-pipeline code base can relatively easily be installed for use in different projects based at the Sanger institute or elsewhere. All that is required is for the Makefile to be modified such that all required software versions, resources and data storage locations are available and valid. The one major complication is that the pipeline code base assumes the presence of a load sharing facility (LSF) and if a different workload management platform were to be used a relatively large number of pipeline scripts within the acgh-pipeline repository would need to be modified.

Perhaps the most important aspect of developing a large scale analytical pipeline is that of data loss prevention. A specific file system structure and database design has been developed to ensure accurate and reliable maintenance of analytical results. The file system is ordered and structured in such a way that missing data or corrupted file formats can be detected using checking scripts that crawl the file system and detect any unexplained data errors. These errors can be reported back as summary emails and correlated with specific data inputs such that any detected data loss (missing data) can be investigated relatively easily. With any large scale analytical process processing errors, potentially resulting in data loss, is almost unavoidable and it is therefore important to have robust safe guards in place to ensure that such situations can be noticed and corrected.

Moreover, the structure and information contained in analytical results should be carefully considered before generating large volume of data. The main goal here is to ensure that the output of an analytical process is both easy to access and to understand for a casual user. This can sometimes be a challenging task where data characteristics are complex and results are hard to interpret. The acgh-pipeline implemented in the DDD project includes a dedicated file system and database assessing package made up of a number of intuitive function calls developed in the R programming language. This has several useful applications; including giving others with little or no knowledge of the data storage structures access to the analytical results with relative ease, abstracting out common code elements that can be used by analytical process elements without the need to rewrite “boiler plate” code, when there is a need to change aspects of the data storage structures all (or most) code changes are restricted to a single package (the data assessor package).

4 | Improved CNV discovery algorithms enable an exon-level resolution map of CNV and reappraisal of the CNV mutation rate.

4.1 OVERVIEW

Copy number variation (CNV) accounts for major differences in DNA sequence between genomes, however its prevalence at the single exon or gene level remains relatively under-categorised. We describe a novel approach for CNV discovery that combines multiple discovery algorithms using dynamic weighting according to the noise properties of the underlying data. Our meta-algorithm, CNsolidate, ranks detections based on differential weighting functions, allowing tuning of Type 1 and Type 2 error rates. We applied this meta-algorithm to 926 apparently healthy controls using exon-targeted array comparative genomic hybridization (exon-CGH), and demonstrate that CNsolidate substantially outperforms individual algorithms. We also present VICAR, a novel Bayesian framework for the classification of CNV inheritance states using SNP genotyping data. Using exon-CGH and single nucleotide polymorphism (SNP) genotyping across 1,766 normal control samples we present a map, including inheritance classifications, for CNVs ranging in size from 44 bp to 8.5 Mb. With a resolution of 15.2 kb we estimate the CNV mutation rate at 1.9×10^{-2} by detecting 16 high confidence rare *de novo* CNVs across 840 parent-offspring transmissions.

4.2 INTRODUCTION

Copy number variants (CNVs) are a major cause of phenotypic variation in humans [182-185] and numerous rare developmental disorders have been associated with specific CNVs affecting dosage sensitive genes or defined syndromic regions [186, 187]. A large number of these genotype-phenotype correlations have been observed in small-scale studies of specific patient groups [188, 189] or by collaborative efforts to share genetic data for rare disease [135]. CNVs have been associated with a number of complex diseases [190, 191]. Studies into schizophrenia [192] and autism [193] have demonstrated the utility of leveraging previously generated single-nucleotide polymorphism (SNP) genotyping data from SNP based genome-wide association studies (GWAS) to search for novel CNV associations.

However, CNV studies have historically been limited by either sample size or technological constraints [13, 50, 104, 194]. Research into the underlying cause of specific clinical conditions has predominantly used array comparative genomic hybridization (array-CGH) as the CNV discovery platform [195] due to its superior sensitivity (dose response) and genomic coverage capability [196]. Yet, the resolution to detect CNVs affecting single genes that is achieved by most commercial and custom array-CGH designs is highly variable due to technical difficulties during probe design [197] or to an insufficient total number of probes available for the array platform. Furthermore, the difficulty and cost of array-CGH experimental protocols generally inhibit its standard application in large-scale research studies [198]. The SNP-based GWAS studies of complex diseases have generally had sufficient sample sizes to search for common CNVs with small effect sizes [199, 200], but SNP genotyping arrays in general have a limited resolution to detect single-exon CNVs due to the distribution of SNPs across the genome [201].

CNV studies can also be limited by the analytical methods used to detect CNVs, and in the availability of parental data, to determine the inheritance states of CNVs. Change point detection algorithms used to call CNVs often produce highly discordant results when applied to the same dataset [128]. Furthermore, when applied to datasets of differing qualities, most algorithms tend to display highly variable results that can sometimes be modeled as a function of the signal-to-noise ratio of the underlying data [202]. Combining information from multiple algorithms can improve CNV calling consistency across datasets [203-205], but most approaches simply use a consensus (naïve voting) approach to combining information from different callers. For example, the CNV detection approach used in the 1000 Genomes Project used an estimated false discovery rate (FDR) rate of greater than 10% to disregard the putative detections least concordant with consensus [36]. Although such approaches can improve CNV detection quality (specificity) compared to single algorithms, the potential for a loss of power (sensitivity) is certainly increased. By requiring a certain number of algorithms to be in agreement for a call to be made (consensus calling) the overall false negative rate (FNR) becomes a function of the union of the FNRs from all callers. Previous studies have inferred inheritance patterns for CNVs detected using SNP genotyping data however, the approach used for inferring

inheritance is often heuristic, for example using a simple call overlap rule [206], or manual, to mitigate the high false negative rate associated with most platforms and analytical pipelines, especially for smaller variants [207].

To facilitate further CNV research, we present an exon-resolution CNV map generated from 926 healthy UK individuals using an exon focused array-CGH platform composed of two 1 million probe Agilent arrays (exon-CGH), and determine the inheritance states of CNVs detected from exon-CGH in 420 offspring using parent-offspring SNP genotyping from a 812K probe Illumina chip. We also describe a software package and framework for data integration developed in the process of generating this map (see **Methods**). These include 1) a change point detection package, CNsolidate, that includes a performance based learning algorithm to define differential weighting functions for multiple algorithms given certain predictive variables drawn from the input data; and 2) a novel automated inheritance classification method, VICAR, which builds on one of the most frequently used software packages for identifying CNVs from SNP data, pennCNV [208]. VICAR is a post-CNV calling tool that uses a Bayesian framework to classify the inheritance of CNVs of interest identified in the exon-CGH offspring data.

CNsolidate is available as **Supplementary Software**.

4.3 METHODS

Samples

The control individuals used here are from two different sample sets from the United Kingdom (UK). The first set of control samples is 565 individuals from the UK blood donor service (UKBS) who gave general consent for anonymised inclusion into genetic studies. The second set of control samples come from the Generation Scotland study and are made up of 420 parent-offspring trios. To ensure that the select offspring are true controls for the DDD study, they were assessed for the presence of suspected learning disabilities as this is a common phenotype in children with undiagnosed developmental disorders. Offspring were eligible for selection if they had measurements that fell within two standard deviations for the mean on four cognitive metrics: Logical Memory from the Wechsler Memory Scale II; Digit Symbol form the Wechsler Adult Intelligence Scale III; Verbal Fluency; Mill Hill Vocabulary Scale. The complete Generation Scotland trios were genotyped using the DDD custom Illumina SNP chip, and the Generation Scotland offspring samples and the UKBS samples were run on the exon-CGH platform.

Array platforms

Exon-CGH Platform: The exon-CGH array is composed of two 1Million probe Agilent arrays and has been heavily targeted towards genes and ultra-conserved elements throughout the human genome. The entire set of GENCODE genes, along with regulatory and mRNA coding elements have been tiled, using a minimum of five oligo-nucleotide probes. The rest of the array content is spent on ensuring the presence of a high-resolution backbone retaining a median probe spacing of 5 kb.

SNP Genotyping Platform: The SNP genotyping array is a customised version of the Illumina Omni-one quad chip. Extra content has been added to standardise the coverage of the array using a ”largest first” gap filling procedure. The gap filling process is aimed at targeting the largest gaps in array coverage first and additionally inserting the best quality probe within the ”central gap region” before moving to the second largest gap in array coverage. In total, the chip includes 811,844 mapped markers (1,734 are unmapped; i.e. are assigned to chromosome only) with a median intermarker distance of 2,378 bp (omitting unmapped markers).

Validation-CGH Platform: The validation aCGH array is the 8 x 60k Agilent format and designed for the validation of exon-CGH calls in 26 samples. It contains 10,000 control probes, randomly selected from the exon-CGH design that are not called in any of the 26 individuals (evenly spread between the 22 autosomes). It also contains 300 CNV tracking probes and 30 gender markers to allow data tracking between the exon-CGH and validation aCGH platforms. The remaining 48,760 probes tile 9,008 CNV regions called across the wscore range.

CNsolidate – Multiple weighted algorithms and expert voting system

See Chapter 2

VICAR - Variant Inheritance Classification Algorithm in R

Model

VICAR (Variant Inheritance Classification Algorithm in R) combines the information from the offspring exon-CGH CNV calling results with Log R ratio (LRR) and B-allele Frequency (BAF) data from SNP genotyping of the complete parent-offspring trios to determine whether a CNV identified from the exon-CGH data is *de novo* or inherited. Given that the purpose of this process is to classify selected, potentially pathogenic CNVs in children with developmental delay, the model is based on the following plausible assumptions: the CNV of interest is rare at a population level; a rare duplication will not occur at the same location as a rare deletion in members of the same family, and vice versa; there are no back-mutations to the copy-neutral state. As the exon-CGH data only indicates the presence of a CNV (and not whether it is homozygous or heterozygous) and the DDD project is primarily focused on rare CNVs, we use a two-state model based on presence or absence of the CNV, rather than considering all five possible states (copy neutral and heterozygous and homozygous deletions and duplications).

If c, f, and m are binary CNV indicators taking value 1 in the presence of a CNV and 0 otherwise, for the offspring, father, and mother respectively; D_{SNP} is the trio SNP data; and D_{aCGH} is the offspring exon-CGH data, then the model can be written as:

$$L(f, m, c|D_{SNP}, D_{aCGH}) \propto P(D_{SNP}|f, m, c) \cdot P(f, m|c) \cdot P(c|D_{aCGH}) \qquad [4\text{-}1]$$

The terms are described in more detail below.

$P(c|D_{aCGH})$ incorporates information from the CNV identification in the offspring via the false-positive rate of the discovery method.

$P(f, m|c)$ is broken down into further constituent terms:

$$P(f, m|c) = \frac{P(c|f, m) \cdot P(f) \cdot P(m)}{P(c)} \qquad [4\text{-}2]$$

$P(c|f, m)$ is the transmission probability for the trio configuration, incorporating a size-specific estimate of the mutation rate. $P(f)$ and $P(m)$ are defined as the probability of randomly selecting a region of the genome of the size of the CNV identified that overlaps a known CNV by at least 50%.

$P(D_{SNP}|f, m, c)$, the joint likelihood of a trio configuration, is based on the likelihoods provided by an external software package; we assume the package used provides estimates for five copy number states (copy neutral and heterozygous and homozygous deletions and duplications) for the genomic region identified in the offspring exon-CGH data. However, as we are using a two-state model, we reduced

the 125 possible five-state trio configurations to the eight configurations needed, combining the likelihoods for each individual in a way that gives appropriate weight to each state contributing to the likelihood. We only consider the possibility of heterozygous CNVs in the parents, because we only consider potentially causal CNVs, but all five states are possible in offspring. The weightings also include the size-specific CNV mutation rate (see **Supplementary Table 2**).

Implementation

VICAR is currently implemented as a package for the R software. Data sources for the model are user-specified; details relating to its use in the DDD project are as follows:

$P(c|D_{aCGH})$ is the false-positive rate associated with the wscore produced by CNsolidate for the CNV identified in the exon-CGH offspring data. We performed a CNV validation experiment on 9008 CNVs spanning the wscore range using a custom designed array-CGH array (Above). We estimated the false positive rate by calculating the proportion of false classifications made between the discovery and validation array at discrete wscore interval.

The size-specific CNV mutation rate incorporated in the terms $P(c|f,m)$ and $P(D_{SNP}|f,m,c)$ is estimated from *details to follow.*

$P(f)$ and $P(m)$ are estimated using the DDD control datasets.

The individual state-specific likelihoods that are incorporated into $P(D_{SNP}|f,m,c)$ are calculated by PennCNV using the LRR and BAF data from the SNP genotyping dataset.

VICAR returns the posterior probabilities for all models and also provides a "hard" classification of inheritance status based on user-specified thresholds relating to (i) declaring the presence of an inherited or *de novo* CNV (default is 0.95); (ii) declaring the absence of a CNV in the offspring (default is 0.99); (iii) the minimum number of probes needed to attempt inheritance classification (default is five probes). Possible classifications are:

- ***de novo***: CNV identified in offspring but not in either parent;
- **paternal**: CNV identified in father and offspring;
- **maternal**: CNV identified in mother and offspring;
- **biparental**: CNV identified in both parents and offspring;
- **no CNV**: no CNV identified in offspring (with or without parental CNV);
- **inconclusive**: This will be returned for two reasons:
    - (i) no model had a posterior probability greater than the threshold for classification
    - (ii) neither *de novo* nor inherited models had posterior probabilities above the classification thresholds;
- **insufficient**: there were less SNPs in the region than the probe threshold.

De novo CNV Parent of Origin Analysis

The origin of each *de novo* CNV was investigated by identifying inheritance-informative positions, sites where one can determine from which parent the child's alleles originated. Such positions include all sites for which parents are opposite homozygotes (e.g. AA and BB), and, on average, half of sites in which one parent is homozygous and the other is heterozygous (e.g. AA and AB). Parents sharing the same genotype are not informative. De novo deletions will result in a single allele remaining in the child, with a B allele frequency of 0, corresponding to the loss of a B allele, or 1, corresponding to the loss of a A allele. De novo duplications will result in an extra allele in the child, with a B allele frequency of 0.33, corresponding to a gain of an A allele, or 0.66, corresponding to a gain of B allele. In five de novo CNV cases, there were no informative sites for which the inheritance could be tracked.

CNV Consensus Reference Set – Defining Common, Rare and Novel CNVs

See Chapter 2.

4.4 RESULTS

Sample cohort

Apparently healthy controls were analysed as part of the Deciphering Developmental Disorders (DDD) project, a UK wide collaboration based at the Wellcome Trust Sanger Institute involving 24 UK and Ireland regional genetics services [29]. We include two different normal control sample sets for CNV detection and inheritance classification: 565 individuals from the UK blood donor service (UKBS) and 420 trios from the Generation Scotland study [209].

CNV discovery algorithm: CNsolidate

We developed a novel detection method, CNsolidate, for CNV discovery using array-CGH data. CNsolidate incorporates 12 independent CNV discovery algorithms and weights each algorithm based on its estimated performance (Type I and Type II error rates) across a range of data noise metrics that are estimated from the input data.

These weights were estimated through extensive simulations of input data with different noise properties. We performed over one million data simulations, varying three specific noise types (see **Methods**) and generated the noise dependent weighting functions, for each algorithm separately, by fitting polynomial curves to the estimated Type I error rate across the given noise value range. CNsolidate calculates a combined CNV confidence measure (wscore) for each copy number variable region (CNVR) detected (see **Methods**). The wscore displays a zero to one range with higher quality calls tending towards a value of one.

CNsolidate Type I and Type II error rates

To estimate the Type I and Type II error rates, using CNsolidate, we generated 73 replicates of the HapMap sample NA12878 on two custom designed Agilent 1 million probe arrays that collectively target each exon with 5 probes and include a backbone of ~700, 000 non-coding probes. We defined true positives (TP) as CNVs that were called in greater than 80% (59) of replicates, and defined CNVs to be the same if they shared greater than a 50% reciprocal overlap (see **Methods**). Overall we defined 12634 TPs above the default wscore threshold of 0.5, 90% of which (11372) were defined as common and had been observed during previous studies at a population frequency greater than one percent (see **Methods**).

We estimated the mean true positive rate (TPR) and mean false positive rate (FPR) for different wscore thresholds (see **Figures 4-1 & 4-2**). We found using a wscore threshold of 0.5 for CNV calls from CNsolidate, results in a TPR of 0.82 and a FPR of 0.052, across all replicates and chose to use 0.5 as our default wscore cut off.

CNsolidate Performance

Compared to its eight (default) component algorithms used individually, CNsolidate displays the best performance in terms of receiver operator curve (ROC) space (see **Figure 4-1**).

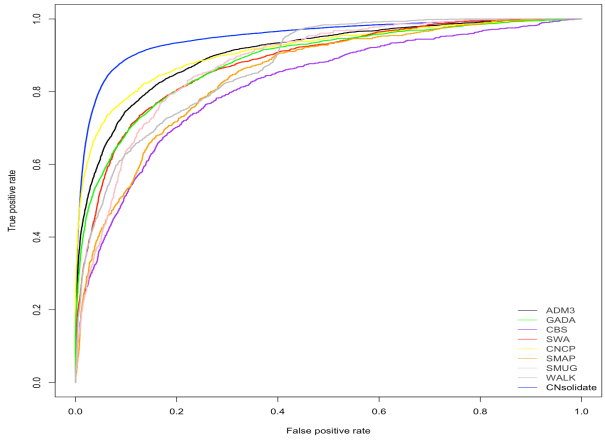


Figure 4-1 Receiver operator curve showing the ROC space using 73 technical replicates for CNsolidate and its eight default algorithms.

There are some differences in the ROC performance of individual algorithms, but no one in particular displays overall poor ROC characteristics indicating that each algorithm has been reasonably well tuned. Furthermore, CNsolidate, using a default wscore threshold of 0.5, achieves an 11% increase in TPR for a 0.05 FPR compared to the highest performing individual algorithm.

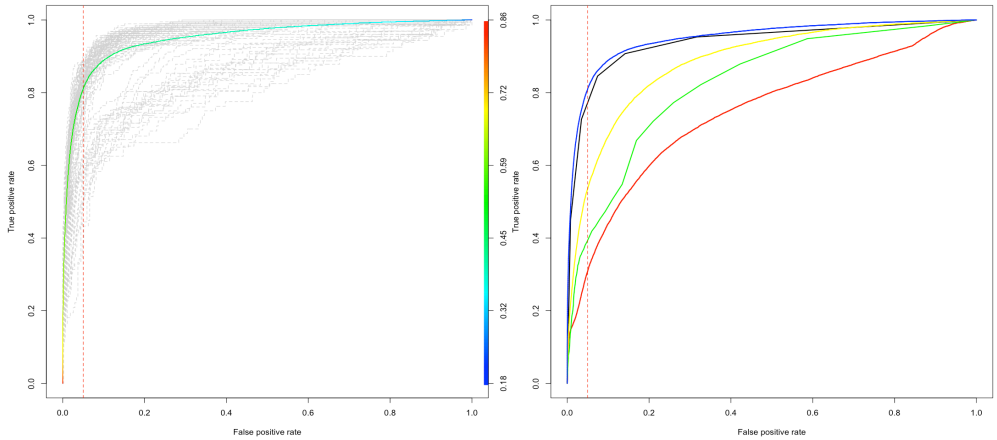


Figure 4-2 Left: Receiver operator curve showing the ROC space using 73 technical replicates and CNsolidate; colored curve is the overall ROC performance colored by the wscore range (color scale on the right hand side of the panel); the ROC performance of individual datasets are shown in grey. Right: Receiver operator curve showing the ROC space using 73 technical replicates and

CNsolidate given different predictors of call quality; the red, green, yellow, black and blue curves show the ROC performance when using the absolute mean log2 ratio, the number of aCGH probes, the p-value, the number of algorithms and the wscore as the call quality predictors respectively.

The wscore is highly negatively correlated with FPR, with increased wscore values decreasing the false positive rate. As the wscore threshold is relaxed, the TPR increases quickly while the FPR increases much more slowly (see **Figure 4-2**). We compared the ROC space observed when using a number of different measures to predict call quality (see **Figure 4-2**). The wscore generated by CNsolidate performed best in terms of ROC space compared to any other measure of call quality that we assessed. The number of algorithms contributing to the detection of a CNV performed second best, however we still observed a significant 0.068 increase in sensitivity when using expert voting (wscore) compared to naïve voting (number of algorithms). The p-value generated by CNsolidate (see **Methods**) provided a reasonable measure of call quality but at a FPR of 0.05 was only able to achieve a TPR of 0.54. We observed that both the number of array-CGH probes and the absolute mean ratio to be poor predictors of call quality overall, having lower than 0.5 TPR with a FPR of 0.05.

CNV validation

We selected 9008 CNVs, spanning the wscore range, detected by CNsolidate in 26 samples, for validation using a custom designed 8x60K CGH array (see **Methods**). We used a similar approach as [50] for validating CNV calls. We calculated the Pearson correlation of the mean log2 ratio values across the 26 samples between the discovery and validation arrays to define a measure of truth. We observed a clear 2-component distribution of correlation values across all CNV calls and applied a nonparametric EM algorithm [210] to determine the mixing proportions for each component (see **Figure 4-3**).

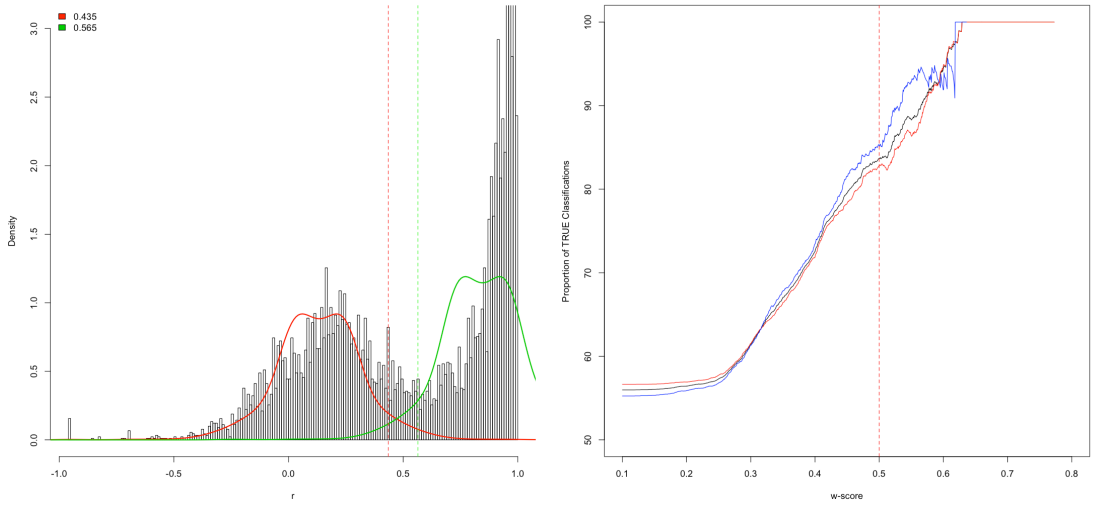


Figure 4-3 Left: Histogram showing the Pearson correlation values for 9,008 CNV between the discovery and validation aCGH platforms. The colored curves show

the result of fitting a nonparametric EM algorithm to the correlation values. The red curve shows the 'false' (not validated) CNV calls and the green curve shows the 'true' (validated) CNV calls. The dashed red and green lines show the mixing proportions for the false and true distributions respectively. Right: A plot showing the proportion of true (validated) CNV calls across the wscore range. The black line show the proportion of true CNV calls across the wscore range overall and the red and blue lines show the proportion of true CNV calls across the wscore range for deletions and duplications respectively.

Correlation values greater than the mean of the mixing proportions (0.5) were used to define true CNV calls, reasoning that the log2 ratio values were sufficiently correlated across all 26 samples between the discovery and validation results. We then calculated the proportion of true (validated) CNV calls across the wscore range for all CNV calls and for losses and gains separately (see **Figure 4-3**). We observed that as the wscore tended towards a value of one the proportion of true classifications sharply increased for both losses and gains. At the default 0.5 wscore threshold the proportion of true classifications was greater than 80% for both losses and gains. Although this could potentially relate to a FPR of 20%, it should be noted that this approach conflates the FPR from the discovery array with the FNR on the validation array. Furthermore, others have shown that reproducibility in replicate experiments on the same array is typically less than 70% for most platforms [202]. As the wscore increased the proportion of true gains increased slightly faster than the proportion of true losses, indicating that that the wscore was slightly better calibrated for predicting gain call quality. For both gains and losses, the proportion of true classifications plateaued at 100% above a wscore value of 0.628.

Algorithm for inferring CNV inheritance: VICAR

For the classification of inheritance states of a predetermined CNV (in this case detected by array-CGH) using SNP genotyping chip data (B-allele frequency and logR ratio) we developed a novel Bayesian framework, Variant Inheritance Classification Algorithm in R (VICAR). VICAR uses the copy number state likelihoods for each trio (or duo) member estimated by PennCNV [208], and combines these with the size and call quality measure for identified CNV, and estimates of the mutation rate and frequency for CNVs of similar size to determine the most likely inheritance configuration. Since the exon-CGH data indicates only the presence of a CNV (but not the number of copies), VICAR currently uses a two-state model based on presence or absence of the CNV, collapsing the five possible copy number states considered by PennCNV. VICAR returns the posterior probabilities for all models, permitting user-defined thresholds to be applied for trio classification (see **Methods**).

Sensitivity of de novo classification

We modeled the sensitivity for *de novo* classifications from VICAR using SNP genotyping data generated on children with previously identified *de novo* CNVs causing developmental disorders (Wessex Regional Genetics Laboratory). Synthetic *de novo* CNVs of sizes 5, 10, 25, 50, 100, 250, 500 and 1000 kb were generated by randomly sampling different numbers of probes from ten large positive control CNVs (eight deletions and two duplications). The performance of

de novo classification using a posterior probability threshold of 0.95 was tested separately for deletions and duplications for regions containing from 2 to 50 SNPs, and for 2 different call qualities (wscore of 0.5 and 0.75). For both deletions and duplications, for both wscore values, the *de novo* classification performance improved as the size and number of SNP markers increased. Overall, the algorithm performed well for *de novo* classification of high confidence CNV calls (as assessed by the wscore) for all sizes and numbers of probes tested, with very small numbers of false negative and inconclusive classifications (see **Figure 4-4**).

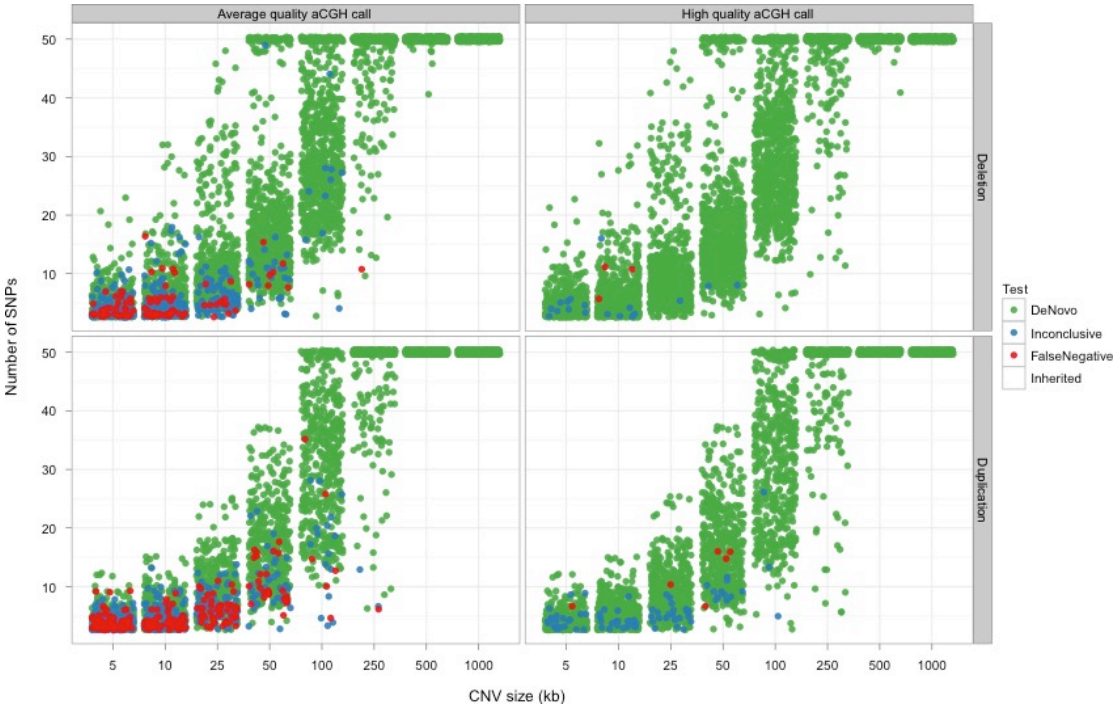


Figure 4-4 Plots showing the number of de novo, inconclusive and false negative classifications from VICAR for deletions and duplications separately, from 5kb to 1Mb, using from 1 to 50 SNP probe markers and for two different wscore values.

For lower confidence array-CGH detections (wscore of 0.5), both the number of SNP markers and the CNV size had a marked effect on the *de novo* classification performance, but sensitivity was good for CNVs with 10 or more markers in the region. As expected, the best predictor for classifier performance was the number of SNP markers, and more deletions could be classified than duplications. Overall we estimated the median resolution for calling *de novo* CNVs as 15.2 kb and 31 kb when using a minimum of five and 10 SNP markers respectively.

CNVs in 926 controls using exon-CGH and CNsolidate

Summary of CNV calls:

We performed CNV discovery using CNsolidate on data generated from 926 control samples on the exon-focused CGH array described above. We used as a reference a pooled DNA made up of 500 control individuals. We called a total of 199,967 autosomal CNVs with a median of 215 per sample. As expected we detected slightly more losses (58%) than gains, it has been well established [208] that gains are more difficult to detect than losses due to a lowered dose response particularly in commonly variable, complex regions of the genome. We defined 35,571/ 199,967 autosomal CNVRs as rare by classifying events with greater than 80% overlap with a CNV of the same type (loss or gain), observed at greater than 1% population frequency (see **Methods**) as being common CNVs. This resulted in classifying 17.8% (17,839 losses and 17,732 gains) of all CNV calls as being rare, indicating that our capacity for calling rare CNVs using CNsolidate was nearly equivalent for losses and gains. The number of autosomal CNV calls detected per sample was very consistent, with a mean of 215 and a 95% probability of being in the interval between 165 and 289 across all 926 exon-CGH datasets (see **Figure 4-5**).

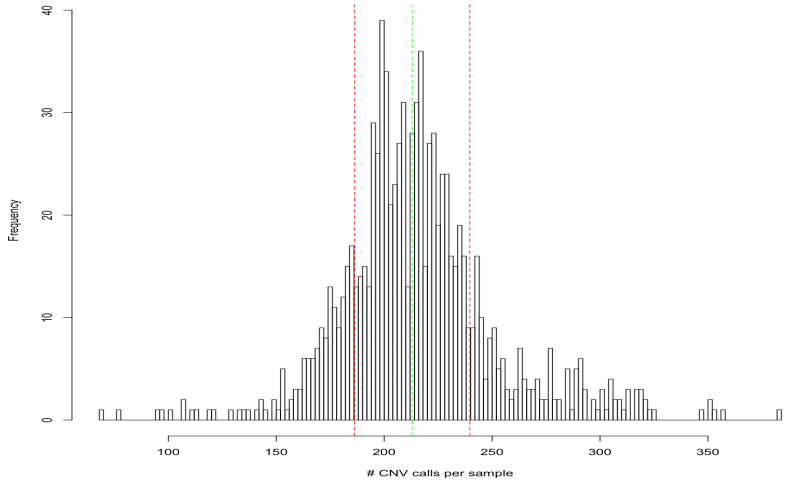


Figure 4-5 Histogram showing the number of autosomal CNV calls made by CNsolidate per sample across 926 exon-CGH data sets. The green line shows the median number of CNV calls and the red lines show the median $\pm$ the median absolute deviation of the number of CNV calls per sample.

The largest CNV detected was 8.5 Mb in length and was observed in two independent samples, one deletion and one duplication sharing precisely the same break points. These large CNVs spanned the entire pericentromeric region on the p arm of chromosome 9, a region well known to contain large common variants [211, 212].

We defined 87,634 (44%) CNVs as single gene events by comparison with the GENCODE gene set (version 17). Of all single gene CNVs 16,695 (19%) included only a single exon with the majority (37%) being intragenic. For all exonic single gene CNVs there was a minimum of 1, median of 3 and maximum of 407 exons per CNV. Additionally we found 4,386 single exon CNVs that were overlapped by multiple genes, with 73%, 26%, 0.8% and 0.02 % of these CNVs being overlapped by 2, 3, 4 and 5 genes respectively. Overall, we detected a total of 21081 single exon CNVs (10% of all detected CNVs), split relatively evenly between losses and gains (11,393 and 9,688 respectively). We defined 3,748 (18%) of all single exon CNVs as rare, a similar rate compared to all CNVs, and defined 1,722 discrete rare single exon CNV loci by collapsing all rare CNV calls using an iterative 50% reciprocal overlap rule. To evaluate the proportion that had been described previously elsewhere we compared the 1,722 rare single exon CNV loci against the database of genomic variation (DGV) [168] using a 50% reciprocal overlap rule. We found that only 14% were contained inside the DGV and add an additional 1,472 discrete rare single exon CNV loci to the literature.

Inheritance classification of CNV Calls in Generation Scotland trios

Using VICAR, we inferred the inheritance status for CNVs detected using exon-CGH in 420 offspring, using SNP-genotyping data from the Generation Scotland trios. To ensure all inheritance classifications were high confidence, we choose to apply a strict cut-off equal to one on the posterior probability of the identified inheritance model estimated by VICAR. Furthermore we only considered CNVs with at least 5 SNP markers to be eligible for inheritance classification; we estimated by simulation that at least five SNP markers were necessary to avoid large numbers of false negative or inconclusive classifications (see **Figure 4**). Overall assigned an informative inheritance classification to 2,792 CNVs ranging in size from 991 bp to 4.8 Mb.

We identified 1,218 paternally inherited, 1,242 maternally inherited, 125 biparental inherited and 177 potential *de novo* CNVs. There were three clear failure modes for *de novo* classifications from VICAR including poor quality SNP genotyping data, uneven SNP probe coverage and the absence of a characteristic BAF split for duplications due to a lack of heterozygous sites within the duplicated segment. After manual review of the 177 *de novo* classifications, 24 were reclassified as paternal, 32 maternal, 15 biparental and 17 as unclear. To further increase the stringency for *de novo* classifications and avoid potential false positives we removed all CNV regions observed to be multi-allelic across the 926 exon-CGH datasets, leaving 16 high confidence *de novo* events ranging in size from 4.9 Kb to 4.8 Mb.

For paternal and maternal classifications we observed a significant increase in the number of informative classifications made for losses compared to gains overall (P=6.26E-09, 7.25E-05 respectively), consistent with the lowered dose response available for gains when using SNP genotyping arrays [213]. For the 16 high confidence *de novo* CNVs we observed no significant difference between the numbers of informative classifications made for losses compared to gains and there was no observable size dependent bias with approximately the same

number of classified *de novo* losses and gains across the genomic size range (see **Figure 4-6**).

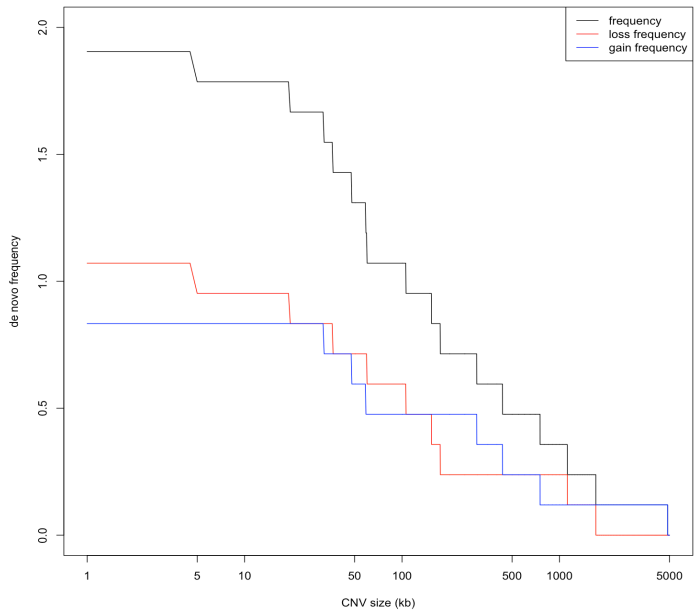


Figure 4-6 The frequency of de novo CNVs against size across 840 parent-offspring transmissions. The black, red and blue lines show the overall, deletion and duplication frequency of de novo CNV events respectively.

Using a similar approach to [207] we determined the parent-of-origin for 15 of 16 high confidence *de novo* CNVs (see **Methods**). We observed a two-fold enrichment for *de novo* CNVs present on the paternally inherited allele, with 10 paternal and five maternal *de novo* events identified (median sizes of 153 kb and 194 kb respectively). Although the sample size was small (N=15) and there was not sufficient statistical evidence for a parent-of-origin effect overall (P = 0.35), the proportion of paternal *de novo* CNVs increased as the size of CNVs decreased (see **Table 4-1**) with 100% of CNVs less than 50 kb in size (N=4) being present on only the paternally inherited allele.

Table 4-1 Table showing the proportion of de novo CNVs present of the paternal allele versus the CNV size in kilobases. The total number of observations with a CNV size less than each size bin is given as N.

CNV Size (kb)	Sample Size (N)	Paternal Proportion
50	4	100%
100	6	83%
250	9	67%
500	11	63%
1000	12	67%
5000	15	67%

We determined that all of the *de novo* CNVs originating from the maternal allele (N=5) had a least one flanking transposable element within a 1 kb window around both CNV breakpoint (start and end) positions. CNVs formed by non-allelic homologous recombination (NAHR) during meiosis are often flanked by low copy repeats (LCR) and transposable elements (TE) [214-216]. Of the *de novo* CNVs originating from the paternal allele (N=10) sixty percent (N=6) were not flanked by LCR or TE supporting the hypothesis [217] that it may be possible to explain the paternal bias for CNV generation by an increase of replication based CNV formation mechanisms such as non-homologous end joining (NHEJ) or microhomology mediated break induced repair (MMBIR) in the male compared to the female germline. We did not observe a significant difference in the paternal age at conception for samples containing a *de novo* CNV on the paternally inherited allele, with a median paternal age of 32 and 30 for samples with paternal *de novo* CNVs flanked and not flanked by a TE respectively compared to a median paternal age of 29 across all samples.

4.5 DISCUSSION

We have developed a novel CNV discovery algorithm, CNsolidate, that integrates and weights CNVs called by up to 12 component algorithms, and we have demonstrated that it has improved performance over single algorithms.

We used CNsolidate to discover CNVs in exon-CGH data from 926 apparently healthy controls. CNV detection, using a default 0.5 wscore cut-off value, was consistent across 926 normal controls and achieved an estimated mean sensitivity of greater than 0.82 with a mean 1-specificity of 0.052. We found that the confidence metric (wscore) generated by CNsolidate was better for predicting call quality compared to any other call quality measure assessed. The estimates for the call quality predictors, absolute mean ratio and number of probes may have been slightly lowered due to conditioning that greater than 80% of replicates must be in agreement for a CNV call to be included in the truth set. For example, it may have been possible to detect additional true CNVs in higher quality datasets that could be distinguished from the background noise in lower quality datasets, or that false low-level CNV calls including many array-CGH probes may have been made in low quality datasets. Nevertheless, the approach used clearly shows an accurate comparison between different call quality measures in providing consistent CNV calls across datasets of varying quality, an important aspect of any large-scale CNV study [104]. In terms of the power to detect genic CNVs that have a higher probability of being biologically meaningful, we observed a clear difference in the number of intronic and exonic CNVs detected per sample when compared to a higher resolution CNV discovery study [50]. We detected on average 40 and 119 intronic and exonic CNVs per sample compared to 269 and 38 from the higher resolution study respectively, reflecting the difference between an exon focused array design (exon-CGH) compared to a genome wide discovery design [50].

We performed a CNV validation experiment using 9008 CNVRs showing that greater than 80% of CNVs above the default wscore cut-off could be experimentally validated. It should be noted that by using the validation data as the gold standard we confound both the false positive detections from the discovery array with the false negative detections from the validation array. Overall, we consider an 80% agreement between results observed from two independent array platforms across 26 samples to be high, [202] showed that reproducibility in replicate experiments on the same array was typically less than 70% for most platforms. Although array-CGH is a relatively mature technology and has been applied to studies of CNV for over a decade [11], it is still possible to make marked advances in the analytical algorithms for both CNV detection and interpretation. As the application of new technologies such as next generation sequencing (NGS) becomes more wide spread in studies of CNV it is important to carefully consider the performance of any available analytical software to ensure high quality CNV discovery.

We also developed a Bayesian approach, VICAR, to determine the inheritance status of a previously discovered CNVs using SNP genotyping chip data from the children and their parents. Using synthetic CNVs of varying sizes derived from

empirical data, we evaluated the capacity of VICAR to infer the inheritance state of CNVs detected from exon-CGH. We found that the most important parameter affecting classifier performance was the number of SNP markers present on the genotyping platform within the CNV region. As expected we observed decreased classification performance for gains compared to losses, likely due to a decreased response of probe intensity for gains compared to losses on array based platforms [218].

CNsolidate is available as a software package with no external dependencies and can be applied to any array-CGH dataset to search for data segments (change point intervals) that are potentially significantly different to the background noise. Although CNsolidate has been primarily developed for use on array-CGH data it can be applied to any time-series like data types, for example read depth data derived from sequence based datasets [219]. The key to maintaining its high performance is the accurate assessment of individual algorithm performances and the creation of reliable algorithm weighting functions. Critical to this approach is ensuring that the predictive variables used to assess individual caller performance are sufficiently descriptive such that all or most differences in data characteristics effecting detection performance are measured. We expect CNsolidate to achieve high performance using its pre-generated algorithm weighting functions however, to achieve optimal performance when applied to different data types, the number, type and ranges of predictive call performance measures need to be carefully considered.

VICAR provides a novel model for incorporating additional information into the classification of inheritance states from parent-offspring data. To apply the model to different data types estimates of the change point call quality, variant mutation rate, variant size and population frequency needs to be defined for the chosen platform. The main limitation for classifying inheritance states during this study was a difference in genomic resolution between the parent (SNP genotyping) and offspring (array-CGH) data, which had a negative impact on the number of CNVs that could be classified with an informative inheritance state.

We have shown an improvement in detection performance when using multiple algorithms to call CNVs from array-CGH data and demonstrated a novel approach to weighting individual algorithm given certain predictive variables (see **Methods**). As the number and performance of analytical software available for calling variants from sequence based data increases [37, 38, 40, 220, 221], it may be worthwhile considering using similar approaches to combining information across multiple callers, improving the detection performance overall.

Across 926 normal control samples, CNsolidate detected a total of 199,967 autosomal calls ranging in size from 44 bp to 8.5 Mb and covering 14.5% of the genome (420 Mb). Using CNsolidate on exon-CGH we were able to detect relatively large numbers of single exon CNVs, with 10% of all CNV calls affecting only a single exon from the GENCODE gene set (version 17). Single exon CNVs have been shown to contribute to a number of phenotypes including severe intellectual disability [222], furthermore the rate at which single exon CNVs are found in normal controls has yet to be fully assessed due to either limited sample

sizes [50], suboptimal genomic resolution [13, 104] or algorithmic complications when detecting single exon CNVs [223]. Here we applied high performance CNV detection using CNsolidate to almost one thousand healthy control samples using exon resolution array-CGH, defining 1,722 discrete rare single exon CNV loci (14% of which are not found in the DGV) and add an additional 1,472 discrete rare single exon CNV loci to current literature. Using SNP genotyping data in 420 parent-offspring trios we obtained an informative inheritance classification for 2,792 CNVs ranging in size from 991 bp to 4.8 Mb. There was no observable bias between the numbers of paternally and maternally inherited CNVs we were able to classify overall. A decrease in power for classifying gains was evident and we observed a significant enrichment in the number of losses fro which we could classify an informative inheritance state. Using highly stringent filtering criteria we defined 16 high confidence *de novo* CNVs and conservatively estimate the CNV mutation rate at 1.9×10^{-2} (16 in 840 transmissions) at a median detection resolution of 15.2 kb.

We estimate a marked increase in the CNV mutation rate compared to a previously published genome-wide study focusing on the identification of large CNVs (>100 kb) across 386 parent‑offspring trios [207]. The authors estimate the CNV mutation rate at 1.2×10^{-2} at a described resolution of approximately 30 kb which is smaller than our estimate of 1.9×10^{-2} at a median resolution of 15.2 kb. Furthermore, 44% (7/16) of the *de novo* CNVs we identified were less than 100 kb in size (see **Figure 4-6**) compared to 22% (2/9) from [207] and for CNVs larger than 500 kb we observed comparable results with a mutation rate of 5×10^{-3} compared to 6.5×10^{-3} [207]. Using exon focused array-CGH (exon-CGH) and SNP genotyping data in 420 parent-offspring trios we were able to classify 16 high confidence *de novo* CNVs at a median resolution of 15.2 kb, suggesting that the CNV mutation rate is likely to be significantly higher than current estimates [224] once the contribution of smaller CNVs is fully assessed. Furthermore, we observed a two-fold enrichment for rare *de novo* CNVs identified on the paternally inherited allele, consistent with the paternal bias previously observed for point mutations [225] and noted that as the size of *de novo* CNVs decreased the proportion of paternal *de novo* CNVs increased (see **Table 4-1**). We determined that 100% (N=5) of maternal *de novo* CNVs were flanked by at least one TE within a 1 kb window around both CNV breakpoint positions compared to only 40% (4/10) of paternal *de novo* CNVs. The difference in CNV mutation rates between the paternally and maternally inherited alleles is likely due to a higher mutation rate from CNV formation mechanisms such as NHEJ and MMBIR compared to NAHR, reflecting the relative difference in mutation rates for replication-based, compared to meiotic-based CNV formation mechanisms [226], in the male opposed to the female germline [207]. The size distribution of paternal *de novo* CNVs with and without flanking TE's showed a marked difference with median sizes of 1 Mb and 53 kb respectively, suggesting that the CNV formation mechanisms of non-recurrent (rare) *de novo* CNV events may show a bias across the size range, with smaller events being preferentially driven by NHEJ and MMBIR compared to NAHR.

5 | Potentially Clinically Relevant Copy Number Variation in 1,012 Normal Control Samples from the Deciphering Developmental Disorders Project.

5.1 OVERVIEW

Copy number variation (CNV) has been a major component of routine clinical medical genetics screening for over a decade. However, the interpretation of individual CNV events remains challenging with most clinical testing laboratories finding potentially pathogenic CNVs in between 10-20% of patients with intellectual disability, autism spectrum disorders, and/or multiple congenital anomalies Furthermore a relatively high proportion of CNVs are normally classified as variants of uncertain significance (VOUS) and laboratories often differ in their classification criteria. These VOUS are generally large, rare CNVs not found in apparently healthy controls however, the selection and reliable usage of a standard common CNV reference set is yet to be established. Applying a rule based approach to CNV filtering in 1012 apparently healthy controls from the Deciphering Developmental Disorders (DDD) project we found CNVs of potential clinical significance in 7.9% and rare de novo CNVs in 2.8% of individuals.

5.2 INTRODUCTION

Clinical microarrays and the analysis of CNVs are well established as a front-line diagnostic test for individuals referred to clinical genetic service laboratories [227]. The field of medical genetics has had major success and seen vast improvements in diagnostic rates due to CNV analysis, linking clinically significant variants in disease causing genes to patient phenotypes across a large number of genomic disorders [228-234]. However, it still remains the minority of patients who receive a potential diagnosis, which is of great importance to patients and families, often influencing clinical treatment, prenatal decision making and genetic counseling [235-237]. During routine CNV analysis in a clinical diagnostic setting only a small proportion of patients (normally less than 20%) receive a definitive or even suggestive genetic diagnosis using clinical microarrays [169]. Furthermore CNVs are often identified which have either unknown clinical significance (variants of uncertain significance) or are in association with a trait completely unrelated to the investigation (incidental finding). This potential for unintended and unexpected findings from clinical microarray data highlights the importance of informed consent and genetic counseling when performing clinical CNV analysis [238]. Large-scale studies such as the International Standards for Cytogenomic Arrays Consortium [239] and the Deciphering Developmental Disorders Project [29] are developing general frameworks and guidelines for the clinical interpretation of microarray and next generation sequence data for patients effected by disorders with a presumed genetic causation [147].

It is likely that the phenotypes displayed for an appreciable number of the patients remaining without a clear diagnosis after clinical microarray screening are due to other types of genetic variation [240-242]. Recent studies using next generation sequence data have found relatively large numbers of undiagnosed patients with *de novo* single nucleotide polymorphisms or insertion / deletion variants in either known or suggestive genomic disorder genes [243, 244]. Even so, due to the difficulty of CNV interpretation and the relatively large number of approaches taken to CNV analysis, it is likely that a significant fraction of undiagnosed patients are due to either poorly understood or poorly captured CNVs. Although the role of *de novo* CNVs in genomic disorders is well characterised, other type of genetic models such as X-linked, recessive, mosaicism, imprinting, digenic or even non-coding CNVs are still to be fully explored. Furthermore, has been observed that the overall burden of CNVs is higher in specific patients groups compared to controls [245, 246] indicating potential combinatorial CNVs effects [247]. Additionally, is conceivable that specific combinations of CNVs, that when observed in isolation would often be ignored, may, by acting in concert, cause extreme phenotypes due to factors such as dosage compensation, incomplete penetrance and polygenic effects [151, 248]. It is clear from initial findings [200, 249] that the sample sizes required to search for such CNVs with small effect sizes will be larger than any currently existing datasets. Most studies to date have used previously generated genome wide association study (GWAS) data however a major limitation yet to be addressed is the availability a normal CNV control data set with both adequate

sample size and effective genomic resolution. With the advance of large-scale genomic initiatives such as the UK10K and GEL projects, as the amount of next generation sequence (NGS) data increases and the analytical methods available to perform CNV analysis from NGS data improve it is likely that these types of CNV effects will become more attainable.

The Deciphering Developmental Disorders (DDD) project is a UK wide collaboration based at the Wellcome Trust Sanger Institute involving the 24 UK and Ireland regional genetics services [29]. We use two different presumed healthy normal control cohorts in the DDD project, 585 individuals from the UK blood donor service (UKBS) and 430 trios from the Generation Scotland study [250]. To highlight the difficulties in interpreting patient CNV data in isolation of detailed patiemt phenotypes we applied an automated rule based CNV filtering system to the DDD control samples with the goal of finding potentially clinically relevant CNVs in apparently healthy individuals.

5.3 METHODS

Overview

We have developed a CNV ranking and prioritisation system in the DDD project for flagging CNVs of potential clinical relevance. CNVs meeting certain criteria relating to potential clinical interest are flagged prior to being reviewed in detail during weekly multidisciplinary reporting meetings.

CNV Consensus Reference Set

To define CNVs as rare we use a common CNV reference set incorporating a number of high quality studies that we term the CNV Consensus. The reference sets that are included in the CNV consensus are; 5919 controls on an Affy6 array from the WTCCC [104], 1892 controls from the 1000 Genomes Project [36], 845 controls on a 2 Million probe Agilent array from the DDD Project [29], 450 controls from a XX prohe Agilent? Array and 40 controls on a 42 Million probe NimbleGen array from the CNV Project [50]. This combined reference set achieves high resolution and adequate sample size, enabling an accurate population frequency estimate for copy number variable events (CNVEs) across the genomic size range (see **Figure 2-27**). We define common CNVEs as those observed at greater than 1% population frequency distinguishing between CNVE type states, CNV loci observed as exclusively loss, exclusively gain or both loss and gain across samples. Patient CNVs sharing greater than 80% of their boundaries with a common CNVE including the same CNV type state are classified as common.

DD Gene to Phenotype Database

The CNV filtering pipeline makes use of a manually curated resource called the DDG2P (dd gene to phenotype) gene list that can be downloaded from the DECIPHER database. The DDG2P is a database of gene to phenotype relationships containing genes with some evidence of an association to a developmental disorder. The database contains gene names, genetic mechanisms, mutation consequences and linked phenotype terms. Each DDG2P entry is placed into one of four possible categories based on the amount of evidence available for the described association. Each gene is associated with specific developmental phenotypes or syndromes via genetic mechanisms and mutation consequences of the gene product. Using DDG2P enables any rare variant in known DD genes with a predictable effect on the gene product to be flagged on the basis of inheritance, genotype and likely mutational consequence.

Flagging CNVs for Clinical Review

We use three different approaches for flagging CNVs of potential clinical interest. First we flag CNVs based on size and rarity alone (VOUS) using a set of fixed CNV size cut-offs dependent on the CNV inheritance and parent affected state. Second we use the DDG2P gene list to search for CNVs in genes previously associated with genomic disorders given the patient gender and parent affected state, the mode and mechanism of each DD gene and the copy number state, genomic location and inheritance classification of each patient CNV. Finally we search for CNVs with a greater than 50% reciprocal overlap with a CNV syndrome region of the same CNV type (losses and gains) contained in the DECIPHER database.

Variants of Uncertain Significance (VOUS)

The class of CNVs flagged using these rules are sufficiently rare in the general population such that they are normally of general clinical interest even if there contribution to patient phenotypes is poorly understood. To flag these types of variants we use a number of fixed cut-offs on rarity and size given the inheritance classification for each detected CNV (see **Table 2-2**). CNVs where the inheritance classification is unknown we use a fixed 500kb cut-off, flagging any rare CNV (loss or gain) greater than 500kb in length for clinical review. For CNVs where we where able to make an informative inheritance classification, if the inheritance is '*de novo*' or biparental the fixed size cut-off used for deletions is 100kb whereas for duplications it is 250kb. Inherited CNVs classified as paternal or maternal can be flagged so long as the father or the mother are affected respectively. Additionally male patients CNVs on chromosome X classified as maternal are flagged irrespective of the mothers affected status.

Developmental Disorder Genes (DDG2P)

The current version of DDG2P includes 1,850 entries (1,336 unique genes) and there are a number of rules needed for flagging CNVs based overlaps with the DDG2P including, sample gender, copy number state of the CNV and the mode and mechanism of the DD gene (see **Table 2-1**).

DD genes with a biallelic mechanism on any chromosomes, in both males and females with either uncertain, loss of function or dominant negative mutational consequences are always flagged if the estimated copy number state is zero. Exactly the same rules are used for monoallelic and X-linked DD genes except that single copy losses (heteroygous deletions) are also flagged. Additionally, gains (copy number greater than 2) in monoallelic DD or X-linked genes with either a uncertain or increased gene dosage mutational consequence are flagged if they encompass the entire gene transcript. CNVs in hemizygous DD genes with either a uncertain, loss of function or dominant negative mutational consequence are flagged if the sample is male and the copy number state is less than two. Gains in hemizygous DD genes CNVs are flagged in both male and female samples if the copy number state is greater than two, the DD gene mutational consequence is either uncertain or increased gene dosage and they encompass the entire gene transcript.

Syndrome Regions (DECIPHER)

To search for events in known disease associated genomic regions we compare rare patients CNVs against a list of known syndromic regions from the DECIPHER database. The DECIPHER syndrome list contains 70 genomic loci, we compared patient losses against regions annotated to be associated to a syndrome when the copy number is less than two and patient gains against regions annotated to be associated to a syndrome when the copy number is greater than two. Patient CNVs we a greater than 50% reciprocal overlap to a DECIPHER syndrome region of the correct CNV type (loss or gain) were flagged for clinical review.

5.4 RESULTS

Variants of potential clinical significance

We previously reported a framework and ranking system for determining variants of potential clinical significance (Wright CF – manuscript under review). There we described a rule-based approach to prioritising CNVs and here we applied exactly the same parameters to the control samples. Overall 67 CNVs in 64/926 samples were flagged for clinical review ranging in size from 223 bp to 8.5 Mb. The majority (78%) were flagged based on size and rarity alone with the remainder (22%) being flagged because they were rare and overlapped a gene previously associated with a known developmental disorder (DD).

Of the 52 CNVs flagged based on rarity and size, 39 were duplications and 13 were deletions suggesting that duplications are better tolerated in the genome than deletions. The Untied Kingdom blood donor service (UKBS) samples, for which SNP genotyping data and consequently inheritance classifications were not available, accounted for 75% of these flagged CNVs highlighting the importance of inheritance information when interpreting large CNVs of uncertain clinical significance. Of the remaining 25%, detected from generation scotland (trio) samples, nine were classified as de novo by VICAR, one was classified as inconclusive due to poor quality SNP genotyping data and three were unclassified due to being on chromosome X (see **Methods**).

The 15 CNVs flagged due to overlapping a DDG2P entry contained a total of 12 unique DD genes. The 8 deletions were all single copy losses and included 6 monoallelic DD genes (4 loss of function and 2 dominant negative). Two genes (SETBP1 & NRXN1) were observed in two independent samples whereas the remaining four (COL9A1, GJB6, AUTS2 & FBN2) were only observed in single samples. The seven duplications included six unique DD genes, three with an uncertain mode of action, one hemizygous loss of function gene and two in genes associated with both developmental disorders and incidental findings. One gene (KCNE1) was observed in two independent samples whereas the remaining five genes (ACADS, GK, MAF, PTCH1 & TP63) were only observed in single samples.

Next we determined that six rare CNVs (three duplications and three deletions) overlapped three known syndromic regions from the DECIPHER database using a greater than 0.5 reciprocal overlap rule. The tree duplications were all classified as inherited. One paternally inherited duplication shared over 95% of its boundaries with the 1q21.1 recurrent microduplication region, a possible susceptibility locus for neurodevelopmental disorders such as autism [251]. The remaining two duplications, one paternally and one maternally inherited, shared over 78% of their boundaries with the 16p13.11 recurrent microduplication region, another neurocognitive disorder susceptibility locus reported to be associated with schizophrenia [252]. The 3 deletions shared over 90% of their boundaries with the steroid sulphatase deficiency (STS) syndrome region on chromosome X. Two of the control samples were female, and both had single copy (heterozygous) deletions encompassing all of the STS syndrome region. The one remaining deletion was homozygous and observed in a male sample, sharing

greater than 99.9% of its boundaries with the STS syndrome region and including the HDHD1,MIR4767,VCX,PNPLA4,MIR651 genes.

5.5 DISCUSSION

We determined the number and type of potentially clinically relevant CNVs detected using exon-CGH in 1,012 control samples by applying a rule-based approach to clinical variant ranking [147]. The majority, 52 CNVs (13 deletions and 39 duplications) were flagged based on rarity and size alone and could collectively be termed variants of uncertain significance (VOUS). However, 15 CNVs (eight deletions and seven duplications) were flagged due to rarity and because they overlapped a known DD gene with the relevant mode and mechanism. These 15 CNVs contained a total of 12 unique DD genes, three were observed in more than one individual whereas nine were only observed once across 1,012 controls. Furthermore, we observed six CNVs sharing greater than 50% positional similarity to a total of four unique previously defined syndrome regions.

Of the recurrently mutated DD genes, two were heterozygous deletions of exon 3 in the SETBP1 gene, a gene associated with Schinzel-Giedion syndrome and Myeloid Leukemia. A previous study [253] described point mutations in 13 individuals with Schinzel-Giedion syndrome and found all mutations to be clustered to a highly conserved 11-bp region on exon 4. Furthermore [254] determined that amino acids 182 to 223 of the Myeloid Leukemia-Associated SET gene, interacted with amino acids 1238 to 1434 of SETBP1. Both CNVs we observe in SETBP1 have not been described in association with DD, delete exon 3, including amino acids 163 to 181, suggesting to us that exon 3 of SETBP1 may be non-essential for gene function. We observed two independent heterozygous deletions of the NRXN1 gene, a monoallelic Schizophrenia susceptibility locus [255] and a biallelic Pitt-Hopkins-like syndrome-2 (PTHSL2) locus [256]. Thereby we report a 0.2% prevalence of Schizophrenia susceptibility due to heterozygous deletions of the NRXN1 gene in normal control individuals. Finally we observed two independent duplications of the KCNE1 gene, both encompassed the entire gene and were not predicted to perturb gene function. The two duplications were flagged due to KCNE1 being an incidental finding gene. KCNE1 is associated with Jervell and Lange-Nielsen syndrome type 2 [257] with a biallelic mode of action and with Long QT syndrome-5 [258] with a monoallelic mode of action. Thus the increased gene dosage conferred by the duplications we observed at KCNE1 was not predicted to predispose the individuals to sudden death from cardiac arrhythmias (severe Long QT syndrome) or cause the phenotypes associated with Jervell and Lange-Nielsen syndrome type 2.

Furthermore six CNVs with sufficient positional similarity to previously defined syndrome regions were flagged. We observed three inherited microduplications at a frequency of one and two in 1,852 transmissions for the 1q21.1 and 16p11.2 syndrome regions respectively. A recent study into the expression of the 1q21.1 microduplication in adults suggested that anticipatory care should include attention to later-onset conditions such as schizophrenia [259]. The 16p11.2 region has been shown by a number of studies to be associated with schizophrenia [252, 260, 261]. We observed three microdeletions at a frequency of three in 1,852 transmissions at the STS syndrome region on chromosome X.

Ichthyosis X-linked (XLI) is a relatively common rare disorder [262], estimated to occur as often as once in 2000 male births [263], we screened 926 normal controls and observed a single homozygous deletion of the STS syndrome region in a male individual. XLI results from steroid sulfatase deficiency and generally only occurs in boys with 90% of patients have deletions of the STS gene. Girls can be carriers but symptoms are usually mild or normal [264, 265]. Most deletions of the STS gene in boys result in the XLI skin disorder and although rare cases of homozygous females from cosanguineous marriages have been reported [266], we assume both female samples here to be phenotypically normal with XLI carrier status. However we do predict the homozygous deletion observed in the male sample to cause the relatively common [263] XLI skin disorder.

Here we report a potential diagnostic feedback rate of 7.9% in normal control samples, 71% of which were rare VOUS, 21% were rare and overlapped a known DD gene and 8% were in defined syndromic regions. Although it might be appropriate, for most CNVs observed in a known DD gene or syndromic region for a patient with a clearly associated phenotype(s) to conclude that the observed phenotype(s) are caused by the specific CNV, it is not necessarily true that the CNVs observed in the known DD gene or syndromic region is sufficient to cause the associated phenotype(s). There are a number of more complex situations that can occur and mask the appearance of a phenotype given the genotype previously associated with DD, such as dosage compensation [149], partial penetrance [150] and polygenic effects [151]. The findings presented here add weight to the general guidelines that genomic findings must be interpreted with a degree of caution and reviewed carefully by clinical experts in relation to patient phenotypes before any solid conclusions should be made.

6 | Discussion & Future Direction

6.1 Discussion

During this research thesis various analytical methodology and software to aid the detection and interpretation of copy number variants (CNVs) from data generated using high-resolution array-CGH has been developed (see **Chapter 2**). A large-scale automated and tightly versioned analytical pipeline for standard use in the Deciphering Developmental Disorders (DDD) project has been implemented (see **Chapter 3**). By applying the methods across over 1,000 apparently healthy control datasets, two specific studies were performed; **a)** A study describing the detection performance achieved by CNsolidate and the overall characteristics of CNVs found using the DDD array-CGH platform (see **Chapter 4**); **b)** A study presenting the number and type of potentially clinically relevant CNVs found across two different control cohorts (see **Chapter 5**).

CNsolidate

CNsolidate is currently in use as the main CNV discovery method for patient data generated as part of a cutting edge translational project based at the Wellcome Trust Sanger Institute (WTSI). This project, the DDD, is a collaboration between the WTSI and the National Health Service (NHS), involving the 24 regional genetic services across the UK and Ireland. The high performance achieved by CNsolidate for CNV detection (see **Chapter 2**), allowing increased detection rates and the accurate ranking of detection quality scores (wscore) has a positive impact on the DDD projects ability to feedback meaningful CNVs to the regional NHS genetic services. The methods developed (CNsolidate) to control the balance between the Type I and Type II error rates across multiple data sets allows a variety of approaches for genomic variation feedback to be taken. The precise definition of high quality (QC passed) CNV calls can be adjusted using a single threshold on the detection quality measure generated by CNsolidate (wscore) across multiple data sets. This allows the DDD array-CGH CNV data to be mined in an iterative manner. Initially, following the normal guidelines for diagnostic testing, the error rates were tuned to achieve high specificity (low Type I error) for an acceptable level of sensitivity (Type II error). Although the DDD project advices that all genetic variation it feeds back should be validated in an accredited laboratory (the regional genetic services), it is important to minimise the reporting of erroneous CNV calls, potentially wasting important resources that could be put to better use. For example, time and money may be better used for additional or complementary assays to aid the search for any underlying genetic causation to specific patient symptoms. In any analysis within a diagnostic setting there is always a careful balance to be made between minimising the chance of reporting an incorrect result (false positive) and maximising the chance of finding the cause of specific patients symptoms (true positive or diagnosis). CNsolidate allows this balance to be modified in a relatively easy and reliable way by simply adjusting the cut-off used on the call quality wscore values, thereby changing the Type I and Type II errors rate balance consistently across large numbers of samples (see **Chapter 2**).

The number of proband (patient) samples expected to have data generated by the DDD array-CGH platform is in excess of 8,000 individuals. It will undoubtedly be the case that a proportion of individuals will receive a potential (suggestive or

compelling) diagnosis based on specific CNVs observed in their genome alone, however unfortunately this will leave a proportion of individuals without any CNV based diagnosis. The expectation is that a number of these 'missed' diagnosis will be due to CNVs that where either not detectable using the DDD array-CGH platform or whose CNVs may have been filtered out due to selecting relatively stringent quality criteria for passing CNVs during clinical reporting. When further mining the DDD patient array-CGH CNV data for potential diagnosis, the wscore (adjusted and non-adjusted) will prove extremely useful for maintaining consistent error rates within and across samples. The DDD is a research project, geared towards furthering the understanding of rare genomic variation and the discovery of novel genetic changes that could potentially be driving certain developmental disorders. Again the mentioned methods inside CNsolidate will allow the search space for potentially pathogenic CNVs to be increased across DDD samples with relative ease, while importantly maintaining a consistent false positive rate.

CNsolidate is not only in use for the DDD project, it is also being trialled by a number of other projects based at the WTSI and elsewhere. The data sets in use for these different projects are rather different to those within the DDD, some being generated using different array platforms and others being generated using completely different sample types, for example aneuploidy (cancer) samples. CNsolidate has been provided to these projects as a standalone piece of detection software and has various predefined default parameter values. CNsolidate is expected to deliver high detection performance using its default settings but if performance is found to be lower than desired specific weighting functions and parameter settings can be tuned using a number of semi-automated simulation based methods within CNsolidate. The performance of CNsolidate for the detection of CNVs from these different projects is yet to be fully assessed but the early indications have all been positive.

Additionally a CNV validation experiment using a custom Agilent array was designed. This array, the validation array, comprises of six 8 x 60k Agilent arrays and the probe content has been spend on tiling CNV detections made with CNsolidate across a number of samples (see **Appendix**). These detections are randomly selected across the detection quality score (wscore) range. This allowed a real false positive rate estimate to be derived across the range of detection quality scores produced by CNsolidate (see **Chapter 4**).

The main message and finding made during the development of CNsolidate is that by using a combination of change point detection algorithms it was possible to increase detection performance compared to individual algorithms. Furthermore, by incorporating some prior knowledge on the performance of individual algorithms given measurable characteristics from the input data, it was possible to improve the ranking of detected segments compared to using a simple consensus based approach. Moreover, by estimating the performance of detection quality ranking between datasets of differing qualities detection ranking measures could be adjusted to achieve a consistent error rate across large numbers of datasets.

Copy Number Tagging SNPs

A novel methodology for the tracking of array-CGH data using CNV tagging SNPs has been developed during this thesis and implemented for the DDD project. This provides some additional confidence that the linked dataset and CNVs potentially fed back to the regional genetic services were detected from the correct sample. Data tracking in the DDD project is extremely important and we have shown that it is possible to use CNV tagging SNPs to detect sample mismatches between a genotyping assay (Sequenom) and an array-CGH experiment. This approach is both considerably more complex and less accurate than the tracking systems used for the sequencing and genotyping analytical pipelines at the WTSI. There are a number of areas that could improve the discriminatory power of the CNV tagging SNPs data sample tracking approach. One obvious improvement would be to increase the number of markers (CNV tagging SNPs) from which to base the probabilities of the model on. Furthermore as the amount of data generated increases the estimates of the model parameters and the definition of copy number state boundaries are likely to improve. This could potentially improve the discriminatory power of the tracking values and lend to a more reliable data tracking system overall.

For the DDD project, due the sensitive nature of the data under consideration, the data tracking method is not used to resolve any sample swaps, rather only to fail sample mismatches and request a new sample. However, we have shown that the approach could potentially be used to resolve such situations (see **Chapter 2**). For example, if by reordering the given sample IDs in the appropriate manner the proportion of sample matches for an entire plate where to increase above a certain level it may be considered as enough evidence to reassign all samples across the plate. Overall the approach developed using CNV tagging SNPs for tracking array-CGH dataset acts as a good proof of principal for array-CGH data tracking as a whole.

CNV Consensus Reference Set

The CNV consensus reference set developed during this thesis is currently used in the DDD project, among others, to allow the accurate filtering of CNVs based on population frequency estimates. It provides positional, frequency and type state information for common CNVEs across a wide genomic size range (see **Chapter 2**). The accurate estimation of population frequency of common CNV is important to variant filtering and allows variants to be classified into different groups based on frequency estimate bins. This is particularly important to any clinical feedback of genomic variation, where it is generally believed that any variant observed above a certain frequency within the general population is unlikely to be pathogenic. Furthermore, this is an important element for various research based questions of potential interest. For example, it may transpire that rare CNVs (those observed with low frequency in the general population) could still contribute to extreme phenotypes. It is not unreasonable to predict that certain rare genomic syndromes may be caused not by a single CNV rather a specific combination of CNVs. Again, it is a reasonable hypothesis that CNVs seen at lower frequency within the general population have a greater potential to be damaging (or contribute to damage) than those seen at the higher end of the frequency spectrum.

The CNV consensus reference set is provided inside CNsolidate along with a number of useful comparison and filtering approaches to allow the accurate and efficient classification of CNVs based on positional, type state and population frequency estimates. By including only studies believed to be high quality and with varying genomic resolutions and samples sizes the CNV consensus achieves both high quality frequency estimates and good genome coverage. Overall the CNV consensus reference set will provide useful functionality to a number of studies into rare and common CNV. It has been made publically available in a variety of different formats and can be downloaded from databases such as DECIPHER. Furthermore, as the amount of genome data increases it may be worth considering adding additional datasets into the CNV consensus or generating additional CNV consensus reference sets. This may be of particular use and interest to studies into CNV using next generation sequencing (NGS) data as currently there are very few reliable common CNV reference sets generated using exclusively NGS data.

CNV Filtering

During this thesis a filtering system for flagging CNVs of potential clinical significance has been developed and implemented for the DDD project. This software relies heavily on a manually curated list of gene to phenotype relationships (DDG2P), the CNV consensus reference set and the call quality measures and copy number estimates provided by CNsolidate. This filtering system is in standard use for the DDD project, facilitating the feedback of potentially clinically relevant CNVs to the clinical genetic teams. The parameters chosen for standard use in the DDD project are aimed at delivering only high confidence, likely causative CNVs for manual review prior to potential deposition in the DECIPHER database and subsequent feedback to the clinical genetic teams. The filtering pipeline has been developed in such a way to allow iterative reporting of CNV calls over time, meaning that as the DDD project progresses it is possible to change filtering parameters and flag additional CNVs for review. This functionality is important not only to allow the filtering parameters to be changed (e.g. relaxed) but also because it is likely that our understanding into the role that CNVs play in developmental disorders will increase over time. This improved understanding may result in new genes being discovered or linked to additional phenotypic characteristics and therefore the filtering of CNVs for potential clinical significance must be repeated in light of this new understanding. Furthermore, technical improvements to the input data, underlying detection (CNsolidate) or classification (CNV consensus) methods may be made and result in a requirement to re-run clinical filtering across all samples within the DDD project. Such a situation was demonstrated (see **Chapter 2**) where a complication with the performance of certain probes on the DDD array-CGH platform was observed and after the solution was implemented the filtering of CNVs for clinical interest needed to be repeated. This highlights the important aspect of reporting potentially relevant genomic variants in general, that, particularly in a diagnostic setting, over time results should always be reassessed by experts in light of any improved knowledge gained.

DDD Pipeline

We have developed a large scale analytical pipeline which is in standard use in the DDD project (see **Chapter 3**). This pipeline is fully versioned and

documented allowing relatively easy maintenance and further development. This is a very important aspect of any software development project but particularly where analytical results may be considered to be of a sensitive nature. As well as providing details of software components and analytical processes, the pipeline documentation allows the pipeline to be executed at scale by anyone with the relevant amount of computational knowledge (e.g. another bioinformatician). This is important to the DDD project generally as maintaining pipeline throughput is important to meet specific project milestones and there should always be a certain degree of redundancy built into any team. Specifically it is beneficial that more than one individual within a team knows how to perform certain analysis such that no one individual becomes irreplaceable.

Drafted Manuscript 1

Chapter 4 is made up of a drafted manuscript that is intended to be submitted to Genome Research. It describes two pieces of analytical software (CNsolidate and VICAR) in detail, presents a number of performance based analysis to demonstrate an improved performance compared to existing methods and describes a high resolution CNV map generated using almost one thousand healthy control samples.

Receiver operator (ROC) analysis using 73 technical replicates was performed comparing the performance of CNsolidate against individual algorithms and comparing CNsolidates call quality measure (wscore) against a number of different call quality predictors. Based on the ROC analysis we show that CNsolidate outperforms all individual algorithms and that the wscore generated by CNsolidate is better at predicting call quality compared to any other quality measures assessed. Additionally a validation experiment is described where a custom validation array was designed targeting CNV calls across the wscore range in 32 selected validation samples (see **Appendix**). By performing an analysis on the correlation in log2 ratio values between the discovery and validation array at 9,008 validation sites across 26 samples it was possible to demonstrate an approximately 80% validation rate at the default wscore cut-off value of 0.5. This rate of validation could be considered high when comparing results across two different platform, others have observed that reproducibility in replicate experiments on the same array is typically less than 70% for most platforms. Additionally an analysis looking at the sensitivity of *de novo* classification achieved by VICAR was performed by using subsampling from a number of positive control (previously known) CNVs. Based on this analysis it was found that the most important parameter affecting classifier performance was the number of SNP markers within the CNV region and there was a decrease in classification performance for gains compared to losses.

For the high resolution CNV map firstly we describe the overall characteristics of CNV calls across the healthy control samples and show a consistent number of total CNV detections per sample. This is a relatively clear indication of high call quality across the sample cohort as large differences in the number of CNVs per genome in healthy individuals is biologically unlikely. Next we show that the number of single exon CNV detections made across the control cohort is relatively large compared to the number available from current literature. When comparing the single exon CNVs detected by CNsolidate against those found within the DGV we report an additional 1,472 single exon CNVs to current

literature. Finally, by using SNP genotyping data in 42- parent-offspring trios, we describe the characteristics of 2,792 informative CNV inheritance classifications made by VICAR. A redefinition and suggested increase in the current CNV mutation rate estimate is made by defining 16 high confidence *de novo* CNVs at a median detection resolution of 15.2 kb across 840 parent offspring transmissions. Furthermore, by performing a parent of origin analysis on the high confidence *de novo* CNVs a marked increase in the number of *de novo* CNVs present on the paternally inherited allele as the size of CNV decreases was evident. This observation reflects the relative difference in mutation rates for replication-based, compared to meiotic-based CNV formation mechanisms in the male opposed to the female germline. Additionally by assessing the number of *de novo* CNVs flanked by transposable elements we suggest that CNV formation mechanisms of non-recurrent (rare) *de novo* CNV events may show a bias across the size range, with smaller events being preferentially driven by NHEJ and MMBIR compared to NAHR.

Drafted Manuscript 2

Chapter 5 is made up of a drafted manuscript that is intended to be submitted to Genome Research. The manuscript is focused on highlighting some of the challenges faced when interpreting genomic data within a clinical setting. Using the CNV clinical filtering pipeline developed in **Chapter 2** we apply filtering to over one thousand healthy control samples to search for potentially clinically relevant CNVs.

Overall a surprisingly high 7.9% potential feedback rate of predicted clinically relevant CNVs in healthy controls is reported. Breaking these flagged CNVs down by type it is clear that the majority (78%) were flagged based on size and rarity alone and could collectively be termed variants of uncertain significance (VOUS). We indicate that 75% of the flagged VOUS CNVs were detected in one of the two healthy control sample cohorts used (the UKBS blood donor cohort). This highlights the importance of having parental data available when interpreting CNVs within a clinical setting. The UKBS cohort did not have parental data available and were therefore enriched for large rare CNV compared to the Generation Scotland cohort. The filtering pipeline excludes CNVs that are classified as inherited from an unaffected parent and as the Generation Scotland cohort had CNV inheritance classifications a proportion of large rare CNVs in child samples would have been filtered out due to an informative parental inheritance classification.

For the remaining flagged CNVs, 15 were flagged due to overlapping a DDG2P entry with the relevant mode and mechanism. Overall, within this class of flagged CNV, there is a relatively low recurrence rate with only 3/15 DDG2P genes being observed in multiple samples. Looking in detail at the characteristics of the recurrently mutated DDG2P genes containing a flagged CNV, for the SETBP1 gene the position of both CNVs is not within the highly conserved 11-bp region on exon 4 or contained within the amino acid sequence predicted to interact with the Myeloid Leukemia-Associated SET gene. Therefore we suggest that the two CNVs affecting exon 3 of SETBP1 observed in healthy controls during this analysis may be non-essential for gene function. Furthermore we suggest that the two flagged duplications found within the KCNE1 gene are unlikely to predispose the individuals to sudden death from cardiac arrhythmias (severe

Long QT syndrome) or cause the phenotypes associated with Jervell and Lange-Nielsen syndrome type 2. A gain in copy number (duplication) of the KCNE1 gene has not been previously associated with either phenotype and the reason the two duplication CNVs were flagged by the filtering pipeline is because KCNE1 is annotated as both a DD and IF gene within the DDG2P database. However, we do report a 0.2% prevalence of Schizophrenia susceptibility due to heterozygous deletions of the NRXN1 gene across the healthy control cohort. Schizophrenia is a complex genomic disorder and we hypothesise that Schizophrenia like phenotypes are potentially relatively under diagnosed in the general population (healthy controls).

Finally we describe 6 CNVs that were flagged due to having sufficient positional similarity with a previously defined syndromic region. Three of these CNVs overlapped with the 1q21.1 and 16p11.2 microduplication regions, both of which have been previously associated with schizophrenia. These three additional CNVs strengthen the hypothesis that complex phenotype characteristics, particularly those affecting mental health such as Schizophrenia, may be present in healthy control cohorts due to poor recognition / under diagnosis within the general population. The 3 remaining flagged CNVs are found at the Ichthyosis skin phenotype linked STS microdeletion syndromic region on chromosome X. Ichthyosis X-linked (XLI) is a relatively common rare disorder estimated to occur as often as once in 2000 male births. Two of the three flagged CNVs were observed in female controls and are therefore predicted to have XLI carrier status. The single CNV at the STS microdeletion region observed in a male sample is predicted to cause the relatively common XLI skin disorder.

Overall the findings presented in **Chapter 5** are aimed at strengthening the general guidelines that interpreting genomic data in a clinical setting should be done with a degree of caution. To avoid the incorrect assignment of variants to phenotypic effects in patient data it is important that clinical, genetic and technical experts work closely together to interpret any biological affect predicted from the analysis of genomic data.

6.2 Future Direction

Moving forward the methods described here are currently being used in large-scale projects such as the DDD. Generally, the overall performance of all these methods across a large-scale study, such as the DDD, acts as a good proof of method robustness. Over the course of the DDD project, the performance of CNsolidate will be under constant assessment and improvements are likely to be made to a number of the underlying models. Furthermore, I would like to create a lightweight version of CNsolidate (CNsolidateLite or similar) for general use by groups interested in change point detection problems.

For CNV interpretation, firstly I would like to add some new data sources into the CNV consensus reference set. As the amount of high quality data sources available for common CNV increases I plan to assess potential data sources and incorporate them where appropriate. I also believe there are some areas of the current CNV filtering approach that could be improved:

1) Currently there is no definition of how gains in copy number (duplications) affect the surrounding genomic landscape. Not all gains in copy number result in an increase in dosage of a gene (or genic element), rather events I term "disruptive duplications" can interrupt gene transcripts (or genic elements) potentially truncating or destroying any gene product. It is a difficult to predict the effect these CNV types might have in terms of any individual genome. However, I expect that the most likely effect of disruptive duplications is to destroy the function of a single gene copy resulting in a similar consequence as for single copy number losses (heterozygous deletions). It is likely that in subsequent versions of the CNV filtering system I will apply a modification to the logic, where I will define disruptive duplications based on their overlap characteristics compared to gene transcripts and treat them in a similar way to single copy losses.

2) Not all genomic disorders are linked to specific genes; rather some are linked to genomic ranges termed syndromic regions. These syndromic regions can be linked to copy number gains or losses or both and have normally been linked to specific phenotypic traits (or syndromes). The DECIPHER database contains seventy defined syndromic regions and in subsequent versions of the CNV filtering system I plan to add these regions as an additional filtering route.

3) Not all phenotypic traits are caused by a single specific CNV; rather some require two or more events in combination due to effects such as dosage compensation. Although currently relatively poorly understood, I expect the number of known "multiple hit" linked loci to see a marked increase in the future. Some of these improvements are likely to be due to specific areas of research within the DDD project and other large scale studies into developmental disorders. I would like to build this functionality into subsequent versions of the CNV filtering system in anticipation of these new exciting research findings.

Finally, although CNVs have been linked to a large number of genomic disorders and has historically been the main focus of diagnostic assays, with the increased usage of NGS data, the number of SNP and INDEL variants linked with patient phenotypes is sure to continue increasing. I see a need to develop a general filtering pipeline for genomic variation of all kinds and to integrate this

information as much as possible. For example a compound heterozygous mutation, resulting in no functional copy of a gene, could result from a single copy loss (CNV) on one allele and a damaging SNP on the other. This is an important complication to resolve and needs to be accounted for to help further the understanding of genomic variation as a whole and its potential effect on the health of an individual.

One major research question that interest me and that I aim to focus on in the future is to what extent common variants (those observed in the general population) contribute individually or in combination to extreme phenotypes. To this goal I plan to carryout a large scale analysis of CNVs detected across the entire DDD cohort (approximately 8,000 patient datasets) once the data generation phase of the DDD array-CGH laboratory has completed.

I believe that the wscore will be especially useful during two main analysis using the DDD array-CGH CNV data **1)** To empower a thorough and detailed assessment of every individual patient in terms of the CNVs detected in their genome using the DDD array-CGH platform. Particularly useful for this analysis will be the non-adjusted wscore as in its raw form (non-adjusted), the wscore provides an accurate ranking of CNV calls in terms of quality across an individual dataset. This is important as the thorough assessment of all CNV calls in any individual genome is a time consuming task requiring a high level of expertise and as such it is important to aid CNV interpretation, decreasing the amount of time an analyst needs to spend on interpreting individual genomes. **2)** To allow different error rates to be explored across datasets in a consistent way using the adjusted wscore when mining the CNV data for novel disease associations. The patients recruited into the DDD study and who have array-CGH CNV calls have also usefully be accurately and consistently phenotyped using the Human Phenotype Ontology (HPO). This will potentially allow groups of patients or even specific patient phenotypes to be associated with specific genetic variants. I expect it will be possible to discover novel CNV changes observed in specific groups of patients sharing particular HPO terms that are not observed (or at lower frequency) across all other patient or control individuals. Key to these types of analysis is maintaining consistent detection error rates across samples as either high false positive rates or low true positive rates in even a few samples within the dataset population can easily lend to false associations and destroy the power of any association test.

Finally, with the vast amount of NGS data being produced by projects such as the DDD and Genomics England (GEL), I would like to test the utility of using CNsolidate, or methods derived from elements of CNsolidate, for the discovery of CNVs using NGS data. Specifically I plan to explore the performance of using a CNsolidate like approach for change detection using the relative difference in read depth across samples as the measure of copy number state.

Bibliography

1. Redon, R., T. Fitzgerald, and N.P. Carter, *Comparative genomic hybridization: DNA labeling, hybridization and detection.* Methods in molecular biology, 2009. 529: p. 267-78.
2. Daniely, M., et al., *Structural unbalanced chromosome rearrangements resolved by comparative genomic hybridization.* Cytogenetics and cell genetics, 1999. 86(1): p. 51-5.
3. Pinkel, D., et al., *High resolution analysis of DNA copy number variation using comparative genomic hybridization to microarrays.* Nature genetics, 1998. 20(2): p. 207-11.
4. Solinas-Toldo, S., et al., *Matrix-based comparative genomic hybridization: biochips to screen for genomic imbalances.* Genes, chromosomes & cancer, 1997. 20(4): p. 399-407.
5. Fiegler, H., R. Redon, and N.P. Carter, *Construction and use of spotted large-insert clone DNA microarrays for the detection of genomic copy number changes.* Nature protocols, 2007. 2(3): p. 577-87.
6. Koolen, D.A., et al., *A novel microdeletion, del(2)(q22.3q23.3) in a mentally retarded patient, detected by array-based comparative genomic hybridization.* Clinical genetics, 2004. 65(5): p. 429-32.
7. Vissers, L.E., et al., *Mutations in a new member of the chromodomain gene family cause CHARGE syndrome.* Nature genetics, 2004. 36(9): p. 955-7.
8. Locke, D.P., et al., *Large-scale variation among human and great ape genomes determined by array comparative genomic hybridization.* Genome research, 2003. 13(3): p. 347-57.
9. Wilson, G.M., et al., *Identification by full-coverage array CGH of human DNA copy number increases relative to chimpanzee and gorilla.* Genome research, 2006. 16(2): p. 173-81.
10. Perry, G.H., et al., *Copy number variation and evolution in humans and chimpanzees.* Genome research, 2008. 18(11): p. 1698-710.
11. Sebat, J., et al., *Large-scale copy number polymorphism in the human genome.* Science, 2004. 305(5683): p. 525-8.
12. Nowak, N.J., et al., *The BAC resource: tools for array CGH and FISH.* Current protocols in human genetics / editorial board, Jonathan L. Haines ... [et al.], 2005. Chapter 4: p. Unit 4 13.
13. Redon, R., et al., *Global variation in copy number in the human genome.* Nature, 2006. 444(7118): p. 444-54.
14. Gu, W., F. Zhang, and J.R. Lupski, *Mechanisms for human genomic rearrangements.* PathoGenetics, 2008. 1(1): p. 4.
15. Stankiewicz, P., et al., *Genomic and genic deletions of the FOX gene cluster on 16q24.1 and inactivating mutations of FOXF1 cause alveolar capillary dysplasia and other malformations.* American journal of human genetics, 2009. 84(6): p. 780-91.
16. Firth, H.V., et al., *DECIPHER: Database of Chromosomal Imbalance and Phenotype in Humans Using Ensembl Resources.* American journal of human genetics, 2009. 84(4): p. 524-33.

17. Barrett, J.C., et al., *Genome-wide association defines more than 30 distinct susceptibility loci for Crohn's disease.* Nature genetics, 2008. 40(8): p. 955-62.
18. Reddy, M.V., et al., *Association between type 1 diabetes and GWAS SNPs in the southeast US Caucasian population.* Genes and immunity, 2011. 12(3): p. 208-12.
19. Zhao, J., et al., *Examination of all type 2 diabetes GWAS loci reveals HHEX-IDE as a locus influencing pediatric BMI.* Diabetes, 2010. 59(3): p. 751-5.
20. Zhang, D., et al., *Accuracy of CNV Detection from GWAS Data.* PloS one, 2011. 6(1): p. e14511.
21. Rancoita, P.M., et al., *An integrated Bayesian analysis of LOH and copy number data.* BMC bioinformatics, 2010. 11: p. 321.
22. King, D.A., et al., *A novel method for detecting uniparental disomy from trio genotypes identifies a significant excess in children with developmental disorders.* Genome research, 2014. 24(4): p. 673-87.
23. Mattes, J., et al., *Paternal uniparental isodisomy for chromosome 14 with mosaicism for a supernumerary marker chromosome 14.* American journal of medical genetics. Part A, 2007. 143A(18): p. 2165-71.
24. McCarroll, S.A., *Extending genome-wide association studies to copy-number variation.* Human molecular genetics, 2008. 17(R2): p. R135-42.
25. Protopopov, A., B. Feng, and L. Chin, *Full complexity genomic hybridization on 60-mer oligonucleotide microarrays for array comparative genomic hybridization (aCGH).* Methods in molecular biology, 2008. 439: p. 87-100.
26. McDonnell, S.K., et al., *Experimental designs for array comparative genomic hybridization technology.* Cytogenetic and genome research, 2013. 139(4): p. 250-7.
27. Warden, M., et al., *Bioinformatics for copy number variation data.* Methods in molecular biology, 2011. 719: p. 235-49.
28. Talseth-Palmer, B.A., et al., *Continuing difficulties in interpreting CNV data: lessons from a genome-wide CNV association study of Australian HNPCC/lynch syndrome patients.* BMC medical genomics, 2013. 6: p. 10.
29. Firth, H.V. and C.F. Wright, *The Deciphering Developmental Disorders (DDD) study.* Developmental medicine and child neurology, 2011. 53(8): p. 702-3.
30. Riggs, E.R., et al., *Towards a Universal Clinical Genomics Database: the 2012 International Standards for Cytogenomic Arrays Consortium Meeting.* Human mutation, 2013. 34(6): p. 915-9.
31. Forer, L., et al., *CONAN: copy number variation analysis software for genome-wide association studies.* BMC bioinformatics, 2010. 11: p. 318.
32. Alonso, A., et al., *CNstream: a method for the identification and genotyping of copy number polymorphisms using Illumina microarrays.* BMC bioinformatics, 2010. 11: p. 264.
33. Wittig, M., et al., *CNVineta: a data mining tool for large case-control copy number variation datasets.* Bioinformatics, 2010. 26(17): p. 2208-9.
34. Wheeler, D.A., et al., *The complete genome of an individual by massively parallel DNA sequencing.* Nature, 2008. 452(7189): p. 872-6.
35. Morozova, O. and M.A. Marra, *Applications of next-generation sequencing technologies in functional genomics.* Genomics, 2008. 92(5): p. 255-64.

36. Abecasis, G.R., et al., *A map of human genome variation from population-scale sequencing.* Nature, 2010. 467(7319): p. 1061-73.
37. Li, H., et al., *The Sequence Alignment/Map format and SAMtools.* Bioinformatics, 2009. 25(16): p. 2078-9.
38. McKenna, A., et al., *The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data.* Genome research, 2010. 20(9): p. 1297-303.
39. Abyzov, A., et al., *CNVnator: an approach to discover, genotype, and characterize typical and atypical CNVs from family and population genome sequencing.* Genome research, 2011. 21(6): p. 974-84.
40. Wei, Z., et al., *SNVer: a statistical tool for variant calling in analysis of pooled or individual next-generation sequencing data.* Nucleic acids research, 2011. 39(19): p. e132.
41. Schmidts, M., et al., *Combined NGS approaches identify mutations in the intraflagellar transport gene IFT140 in skeletal ciliopathies with early progressive kidney Disease.* Human mutation, 2013. 34(5): p. 714-24.
42. Raffan, E., et al., *Early Diagnosis of Werner's Syndrome Using Exome-Wide Sequencing in a Single, Atypical Patient.* Frontiers in endocrinology, 2011. 2: p. 8.
43. Sherry, S.T., et al., *dbSNP: the NCBI database of genetic variation.* Nucleic acids research, 2001. 29(1): p. 308-11.
44. Thusberg, J., A. Olatubosun, and M. Vihinen, *Performance of mutation pathogenicity prediction methods on missense variants.* Human mutation, 2011. 32(4): p. 358-68.
45. Harakalova, M., et al., *Genomic DNA pooling strategy for next-generation sequencing-based rare variant discovery in abdominal aortic aneurysm regions of interest-challenges and limitations.* Journal of cardiovascular translational research, 2011. 4(3): p. 271-80.
46. Huber, J., et al., *Molecular Screening for 22Q11.2 Deletion Syndrome in Patients With Congenital Heart Disease.* Pediatric cardiology, 2014.
47. Lander, E.S., et al., *Initial sequencing and analysis of the human genome.* Nature, 2001. 409(6822): p. 860-921.
48. Postier, B., et al., *Benefits of in-situ synthesized microarrays for analysis of gene expression in understudied microorganisms.* Journal of microbiological methods, 2008. 74(1): p. 26-32.
49. Haas, S.A., et al., *Genome-scale design of PCR primers and long oligomers for DNA microarrays.* Nucleic acids research, 2003. 31(19): p. 5576-81.
50. Conrad, D.F., et al., *Origins and functional impact of copy number variation in the human genome.* Nature, 2010. 464(7289): p. 704-12.
51. Fiegler, H., et al., *DNA microarrays for comparative genomic hybridization based on DOP-PCR amplification of BAC and PAC clones.* Genes, chromosomes & cancer, 2003. 36(4): p. 361-74.
52. Altman, N., *Replication, variation and normalisation in microarray experiments.* Applied bioinformatics, 2005. 4(1): p. 33-44.
53. Hester, S.D., et al., *Comparison of comparative genomic hybridization technologies across microarray platforms.* Journal of biomolecular techniques : JBT, 2009. 20(2): p. 135-51.

54. Fiegler, H., et al., *Accurate and reliable high-throughput detection of copy number variation in the human genome.* Genome research, 2006. 16(12): p. 1566-74.
55. Sellick, G.S., et al., *Genomewide linkage searches for Mendelian disease loci can be efficiently conducted using high-density SNP genotyping arrays.* Nucleic acids research, 2004. 32(20): p. e164.
56. Shi, J. and P. Li, *An integrative segmentation method for detecting germline copy number variations in SNP arrays.* Genetic epidemiology, 2012. 36(4): p. 373-83.
57. Tucker, T., et al., *Comparison of genome-wide array genomic hybridization platforms for the detection of copy number variants in idiopathic mental retardation.* BMC medical genomics, 2011. 4: p. 25.
58. Haraksingh, R.R., et al., *Genome-wide mapping of copy number variation in humans: comparative analysis of high resolution array platforms.* PloS one, 2011. 6(11): p. e27859.
59. Koumi, P., et al., *Evaluation and validation of the ABI 3700, ABI 3100, and the MegaBACE 1000 capillary array electrophoresis instruments for use with short tandem repeat microsatellite typing in a forensic environment.* Electrophoresis, 2004. 25(14): p. 2227-41.
60. Yeung, S.H., et al., *Rapid and high-throughput forensic short tandem repeat typing using a 96-lane microfabricated capillary array electrophoresis microdevice.* Journal of forensic sciences, 2006. 51(4): p. 740-7.
61. McPherson, J.D., et al., *A physical map of the human genome.* Nature, 2001. 409(6822): p. 934-41.
62. Sanger, F., S. Nicklen, and A.R. Coulson, *DNA sequencing with chain-terminating inhibitors.* Proceedings of the National Academy of Sciences of the United States of America, 1977. 74(12): p. 5463-7.
63. Bentley, D.R., et al., *Accurate whole human genome sequencing using reversible terminator chemistry.* Nature, 2008. 456(7218): p. 53-9.
64. Margulies, M., et al., *Genome sequencing in microfabricated high-density picolitre reactors.* Nature, 2005. 437(7057): p. 376-80.
65. Schuster, S.C., *Next-generation sequencing transforms today's biology.* Nature methods, 2008. 5(1): p. 16-8.
66. Porreca, G.J., *Genome sequencing on nanoballs.* Nature biotechnology, 2010. 28(1): p. 43-4.
67. Ribeiro, F.J., et al., *Finished bacterial genomes from shotgun sequence data.* Genome research, 2012. 22(11): p. 2270-7.
68. Bashir, A., et al., *A hybrid approach for the automated finishing of bacterial genomes.* Nature biotechnology, 2012. 30(7): p. 701-7.
69. Koren, S., et al., *Hybrid error correction and de novo assembly of single-molecule sequencing reads.* Nature biotechnology, 2012. 30(7): p. 693-700.
70. Lo, C., et al., *Strobe sequence design for haplotype assembly.* BMC bioinformatics, 2011. 12 Suppl 1: p. S24.
71. Bao, S., et al., *Evaluation of next-generation sequencing software in mapping and assembly.* Journal of human genetics, 2011. 56(6): p. 406-14.
72. Lindblom, A. and P.N. Robinson, *Bioinformatics for human genetics: promises and challenges.* Human mutation, 2011. 32(5): p. 495-500.

73. Ji, Y., et al., *BM-map: Bayesian mapping of multireads for next-generation sequencing data.* Biometrics, 2011. 67(4): p. 1215-24.
74. Talkowski, M.E., et al., *Next-generation sequencing strategies enable routine detection of balanced chromosome rearrangements for clinical diagnostics and genetic research.* American journal of human genetics, 2011. 88(4): p. 469-81.
75. Komura, D., et al., *Genome-wide detection of human copy number variations using high-density DNA oligonucleotide arrays.* Genome research, 2006. 16(12): p. 1575-84.
76. Peiffer, D.A., et al., *High-resolution genomic profiling of chromosomal aberrations using Infinium whole-genome genotyping.* Genome research, 2006. 16(9): p. 1136-48.
77. Shalon, D., S.J. Smith, and P.O. Brown, *A DNA microarray system for analyzing complex DNA samples using two-color fluorescent probe hybridization.* Genome research, 1996. 6(7): p. 639-45.
78. Lassmann, S., et al., *Array CGH identifies distinct DNA copy number profiles of oncogenes and tumor suppressor genes in chromosomal- and microsatellite-unstable sporadic colorectal carcinomas.* Journal of molecular medicine, 2007. 85(3): p. 293-304.
79. Korbel, J.O., et al., *Paired-end mapping reveals extensive structural variation in the human genome.* Science, 2007. 318(5849): p. 420-6.
80. Plagnol, V., et al., *A robust model for read count data in exome sequencing experiments and implications for copy number variant calling.* Bioinformatics, 2012. 28(21): p. 2747-54.
81. Zhang, Z.D. and M.B. Gerstein, *Detection of copy number variation from array intensity and sequencing read depth using a stepwise Bayesian model.* BMC bioinformatics, 2010. 11: p. 539.
82. Chari, R., W.W. Lockwood, and W.L. Lam, *Computational methods for the analysis of array comparative genomic hybridization.* Cancer informatics, 2006. 2: p. 48-58.
83. Yang, Y.H., et al., *Normalization for cDNA microarray data: a robust composite method addressing single and multiple slide systematic variation.* Nucleic acids research, 2002. 30(4): p. e15.
84. Staaf, J., et al., *Normalization of array-CGH data: influence of copy number imbalances.* BMC genomics, 2007. 8: p. 382.
85. Wang, J., J.Z. Ma, and M.D. Li, *Normalization of cDNA microarray data using wavelet regressions.* Combinatorial chemistry & high throughput screening, 2004. 7(8): p. 783-91.
86. Skvortsov, D., et al., *Using expression arrays for copy number detection: an example from E. coli.* BMC bioinformatics, 2007. 8: p. 203.
87. Workman, C., et al., *A new non-linear normalization method for reducing variability in DNA microarray experiments.* Genome biology, 2002. 3(9): p. research0048.
88. Springer, N.M., et al., *Maize inbreds exhibit high levels of copy number variation (CNV) and presence/absence variation (PAV) in genome content.* PLoS genetics, 2009. 5(11): p. e1000734.
89. Fitzgerald, T.W., et al., *aCGH.Spline--an R package for aCGH dye bias normalization.* Bioinformatics, 2011. 27(9): p. 1195-200.

90. Fang, H., et al., *Self-self hybridization as an alternative experiment design to dye swap for two-color microarrays.* Omics : a journal of integrative biology, 2007. 11(1): p. 14-24.
91. Koren, A., I. Tirosh, and N. Barkai, *Autocorrelation analysis reveals widespread spatial biases in microarray experiments.* BMC genomics, 2007. 8: p. 164.
92. Marioni, J.C., et al., *Breaking the waves: improved detection of copy number variation from microarray-based comparative genomic hybridization.* Genome biology, 2007. 8(10): p. R228.
93. Diskin, S.J., et al., *Adjustment of genomic waves in signal intensities from whole-genome SNP genotyping platforms.* Nucleic acids research, 2008. 36(19): p. e126.
94. Lepretre, F., et al., *Waved aCGH: to smooth or not to smooth.* Nucleic acids research, 2010. 38(7): p. e94.
95. Coifman, R.R. and M.V. Wickerhauser, *Wavelets, adapted waveforms and de-noising.* Electroencephalography and clinical neurophysiology. Supplement, 1996. 45: p. 57-78.
96. Gopalappa, C., et al., *Removal of hybridization and scanning noise from microarrays.* IEEE transactions on nanobioscience, 2009. 8(3): p. 210-8.
97. Wang, Y. and S. Wang, *A novel stationary wavelet denoising algorithm for array-based DNA Copy Number data.* International journal of bioinformatics research and applications, 2007. 3(2): p. 206-22.
98. Chang, W.R. and I.P. McLean, *CUSUM: a tool for early feedback about performance?* BMC medical research methodology, 2006. 6: p. 8.
99. Syvanen, A.C., *Accessing genetic variation: genotyping single nucleotide polymorphisms.* Nature reviews. Genetics, 2001. 2(12): p. 930-42.
100. Adler, A.J., G.B. Wiley, and P.M. Gaffney, *Infinium assay for large-scale SNP genotyping applications.* Journal of visualized experiments : JoVE, 2013(81): p. e50683.
101. LaFramboise, T., *Single nucleotide polymorphism arrays: a decade of biological, computational and technological advances.* Nucleic acids research, 2009. 37(13): p. 4181-93.
102. Sim, S.C., et al., *Development of a large SNP genotyping array and generation of high-density genetic maps in tomato.* PloS one, 2012. 7(7): p. e40563.
103. Barnes, C., et al., *A robust statistical method for case-control association testing with copy number variation.* Nature genetics, 2008. 40(10): p. 1245-52.
104. Craddock, N., et al., *Genome-wide association study of CNVs in 16,000 cases of eight common diseases and 3,000 shared controls.* Nature, 2010. 464(7289): p. 713-20.
105. Pearson, T.A. and T.A. Manolio, *How to interpret a genome-wide association study.* JAMA : the journal of the American Medical Association, 2008. 299(11): p. 1335-44.
106. Tang, H., et al., *Genetic structure, self-identified race/ethnicity, and confounding in case-control association studies.* American journal of human genetics, 2005. 76(2): p. 268-75.
107. Marshall, T., *What is a case-control study?* International journal of epidemiology, 2004. 33(3): p. 612-3.

108. Hintsanen, P., et al., *An empirical comparison of case-control and trio based study designs in high throughput association mapping.* Journal of medical genetics, 2006. 43(7): p. 617-24.
109. Serratosa, J.M., et al., *Clinical and genetic analysis of a large pedigree with juvenile myoclonic epilepsy.* Annals of neurology, 1996. 39(2): p. 187-95.
110. Wang, C.C., et al., *Gene dosage imbalance of human chromosome 21 in mouse embryonic stem cells differentiating to neurons.* Gene, 2011. 481(2): p. 93-101.
111. Henrichsen, C.N., E. Chaignat, and A. Reymond, *Copy number variants, diseases and gene expression.* Human molecular genetics, 2009. 18(R1): p. R1-8.
112. Veitia, R.A. and J.A. Birchler, *Dominance and gene dosage balance in health and disease: why levels matter!* The Journal of pathology, 2010. 220(2): p. 174-85.
113. Anneren, G. and B. Edman, *Down syndrome--a gene dosage disease caused by trisomy of genes within a small segment of the long arm of chromosome 21, exemplified by the study of effects from the superoxide-dismutase type 1 (SOD-1) gene.* APMIS. Supplementum, 1993. 40: p. 71-9.
114. Geeleher, P., N.J. Cox, and R.S. Huang, *Clinical drug response can be predicted using baseline gene expression levels and in vitro drug sensitivity in cell lines.* Genome biology, 2014. 15(3): p. R47.
115. Ansley, S.J., et al., *Basal body dysfunction is a likely cause of pleiotropic Bardet-Biedl syndrome.* Nature, 2003. 425(6958): p. 628-33.
116. Cheng, Y.W., et al., *Copy number analysis of NIPBL in a cohort of 510 patients reveals rare copy number variants and a mosaic deletion.* Molecular genetics & genomic medicine, 2014. 2(2): p. 115-23.
117. Amir, R.E., et al., *Rett syndrome is caused by mutations in X-linked MECP2, encoding methyl-CpG-binding protein 2.* Nature genetics, 1999. 23(2): p. 185-8.
118. Garber, K.B., J. Visootsak, and S.T. Warren, *Fragile X syndrome.* European journal of human genetics : EJHG, 2008. 16(6): p. 666-72.
119. Pereira, P.M., et al., *Coffin-Lowry syndrome.* European journal of human genetics : EJHG, 2010. 18(6): p. 627-33.
120. Biggar, W.D., et al., *Duchenne muscular dystrophy: current knowledge, treatment, and future prospects.* Clinical orthopaedics and related research, 2002(401): p. 88-106.
121. Wraith, J.E., et al., *Mucopolysaccharidosis type II (Hunter syndrome): a clinical review and recommendations for treatment in the era of enzyme replacement therapy.* European journal of pediatrics, 2008. 167(3): p. 267-77.
122. Lindsay, A.C., J.M. Behar, and J. McEwan, *Images in cardiology. Fabry's disease, an X-linked recessive condition, can have isolated cardiac manifestations in heterozygote females.* Heart, 2006. 92(5): p. 685.
123. Sacconi, S., L. Salviati, and C. Desnuelle, *Facioscapulohumeral muscular dystrophy.* Biochimica et biophysica acta, 2014.
124. Ferguson-Smith, M.A. and D.R. Goudie, *Digenic/multilocus aetiology of multiple self-healing squamous epithelioma (Ferguson-Smith disease): TGFBR1 and a second linked locus.* The international journal of biochemistry & cell biology, 2014.

125. Schaffer, A.A., *Digenic inheritance in medical genetics.* Journal of medical genetics, 2013. 50(10): p. 641-52.
126. Lindhurst, M.J., et al., *A mosaic activating mutation in AKT1 associated with the Proteus syndrome.* The New England journal of medicine, 2011. 365(7): p. 611-9.
127. Dellinger, A.E., et al., *Comparative analyses of seven algorithms for copy number variant identification from single nucleotide polymorphism arrays.* Nucleic acids research, 2010. 38(9): p. e105.
128. Roy, S. and A. Motsinger Reif, *Evaluation of calling algorithms for array-CGH.* Frontiers in genetics, 2013. 4: p. 217.
129. Wineinger, N.E., et al., *Statistical issues in the analysis of DNA Copy Number Variations.* International journal of computational biology and drug design, 2008. 1(4): p. 368-95.
130. Bhagwat, M., *Searching NCBI's dbSNP database.* Current protocols in bioinformatics / editoral board, Andreas D. Baxevanis ... [et al.], 2010. Chapter 1: p. Unit 1 19.
131. Lappalainen, I., et al., *DbVar and DGVa: public archives for genomic structural variation.* Nucleic acids research, 2013. 41(Database issue): p. D936-41.
132. Rosenfeld, J.A., C.E. Mason, and T.M. Smith, *Limitations of the human reference genome for personalized genomics.* PloS one, 2012. 7(7): p. e40294.
133. Valsesia, A., et al., *The Growing Importance of CNVs: New Insights for Detection and Clinical Interpretation.* Frontiers in genetics, 2013. 4: p. 92.
134. Hehir-Kwa, J.Y., et al., *Pathogenic or not? Assessing the clinical relevance of copy number variants.* Clinical genetics, 2013. 84(5): p. 415-21.
135. Swaminathan, G.J., et al., *DECIPHER: web-based, community resource for clinical interpretation of rare variants in developmental disorders.* Human molecular genetics, 2012. 21(R1): p. R37-44.
136. Bragin, E., et al., *DECIPHER: database for the interpretation of phenotype-linked plausibly pathogenic sequence and copy-number variation.* Nucleic acids research, 2014. 42(Database issue): p. D993-D1000.
137. Corpas, M., et al., *Interpretation of genomic copy number variants using DECIPHER.* Current protocols in human genetics / editorial board, Jonathan L. Haines ... [et al.], 2012. Chapter 8: p. Unit 8 14.
138. Gabriel, S., L. Ziaugra, and D. Tabbaa, *SNP genotyping using the Sequenom MassARRAY iPLEX platform.* Current protocols in human genetics / editorial board, Jonathan L. Haines ... [et al.], 2009. Chapter 2: p. Unit 2 12.
139. Harrow, J., et al., *GENCODE: the reference human genome annotation for The ENCODE Project.* Genome research, 2012. 22(9): p. 1760-74.
140. Pique-Regi, R., A. Caceres, and J.R. Gonzalez, *R-Gada: a fast and flexible pipeline for copy number analysis in association studies.* BMC bioinformatics, 2010. 11: p. 380.
141. Olshen, A.B., et al., *Circular binary segmentation for the analysis of array-based DNA copy number data.* Biostatistics, 2004. 5(4): p. 557-72.
142. Price, T.S., et al., *SW-ARRAY: a dynamic programming solution for the identification of copy-number changes in genomic DNA using array comparative genome hybridization data.* Nucleic acids research, 2005. 33(11): p. 3455-64.

143. Benelli, M., et al., *A very fast and accurate method for calling aberrations in array-CGH data.* Biostatistics, 2010. 11(3): p. 515-8.
144. Andersson, R., et al., *A segmental maximum a posteriori approach to genome-wide copy number profiling.* Bioinformatics, 2008. 24(6): p. 751-8.
145. Erdman, C. and J.W. Emerson, *A fast Bayesian change point analysis for the segmentation of microarray data.* Bioinformatics, 2008. 24(19): p. 2143-8.
146. Taylor, J., et al., *Using galaxy to perform large-scale interactive data analyses.* Current protocols in bioinformatics / editoral board, Andreas D. Baxevanis ... [et al.], 2007. Chapter 10: p. Unit 10 5.
147. Wright, C.F., et al., *Genetic diagnosis of developmental disorders in the DDD study: a scalable analysis of genome-wide research data.* Lancet, 2014.
148. Ballif, B.C., et al., *High-resolution array CGH defines critical regions and candidate genes for microcephaly, abnormalities of the corpus callosum, and seizure phenotypes in patients with microdeletions of 1q43q44.* Human genetics, 2012. 131(1): p. 145-56.
149. Prestel, M., C. Feller, and P.B. Becker, *Dosage compensation and the global re-balancing of aneuploid genomes.* Genome biology, 2010. 11(8): p. 216.
150. Girirajan, S., et al., *Phenotypic heterogeneity of genomic disorders and rare copy-number variants.* The New England journal of medicine, 2012. 367(14): p. 1321-31.
151. Zahnleiter, D., et al., *Rare copy number variants are a common cause of short stature.* PLoS genetics, 2013. 9(3): p. e1003365.
152. Kohler, S., et al., *The Human Phenotype Ontology project: linking molecular biology and disease through phenotype data.* Nucleic acids research, 2014. 42(Database issue): p. D966-74.
153. Kohler, S., et al., *Ontological phenotype standards for neurogenetics.* Human mutation, 2012. 33(9): p. 1333-9.
154. Robinson, P.N. and S. Mundlos, *The human phenotype ontology.* Clinical genetics, 2010. 77(6): p. 525-34.
155. Robinson, P.N., et al., *The Human Phenotype Ontology: a tool for annotating and analyzing human hereditary disease.* American journal of human genetics, 2008. 83(5): p. 610-5.
156. Bellenguez, C., et al., *A robust clustering algorithm for identifying problematic samples in genome-wide association studies.* Bioinformatics, 2012. 28(1): p. 134-5.
157. Clark, A.G., et al., *Ascertainment bias in studies of human genome-wide polymorphism.* Genome research, 2005. 15(11): p. 1496-502.
158. Goldstein, D.B. and G.L. Cavalleri, *Genomics: understanding human diversity.* Nature, 2005. 437(7063): p. 1241-2.
159. Hinds, D.A., et al., *Whole-genome patterns of common DNA variation in three human populations.* Science, 2005. 307(5712): p. 1072-9.
160. Myers, S., et al., *A fine-scale map of recombination rates and hotspots across the human genome.* Science, 2005. 310(5746): p. 321-4.
161. Daly, M.J., et al., *High-resolution haplotype structure in the human genome.* Nature genetics, 2001. 29(2): p. 229-32.
162. Duitama, J., et al., *Fosmid-based whole genome haplotyping of a HapMap trio child: evaluation of Single Individual Haplotyping techniques.* Nucleic acids research, 2012. 40(5): p. 2041-53.

163. Flicek, P., et al., *Ensembl 2014.* Nucleic acids research, 2014. 42(Database issue): p. D749-55.
164. Flicek, P., et al., *Ensembl 2013.* Nucleic acids research, 2013. 41(Database issue): p. D48-55.
165. Flicek, P., et al., *Ensembl 2012.* Nucleic acids research, 2012. 40(Database issue): p. D84-90.
166. Flicek, P., et al., *Ensembl 2011.* Nucleic acids research, 2011. 39(Database issue): p. D800-6.
167. Karolchik, D., et al., *The UCSC Genome Browser database: 2014 update.* Nucleic acids research, 2014. 42(Database issue): p. D764-70.
168. MacDonald, J.R., et al., *The Database of Genomic Variants: a curated collection of structural variation in the human genome.* Nucleic acids research, 2014. 42(Database issue): p. D986-92.
169. Peters, G.B. and M.D. Pertile, *Chromosome microarrays in diagnostic testing: interpreting the genomic data.* Methods in molecular biology, 2014. 1168: p. 117-55.
170. Cox, C., et al., *A survey of homozygous deletions in human cancer genomes.* Proceedings of the National Academy of Sciences of the United States of America, 2005. 102(12): p. 4542-7.
171. Lim, E.T., et al., *Rare complete knockouts in humans: population distribution and significant role in autism spectrum disorders.* Neuron, 2013. 77(2): p. 235-42.
172. Van Schil, K., et al., *Early-onset autosomal recessive cerebellar ataxia associated with retinal dystrophy: new human hotfoot phenotype caused by homozygous GRID2 deletion.* Genetics in medicine : official journal of the American College of Medical Genetics, 2014.
173. Mozzillo, E., et al., *A novel CISD2 intragenic deletion, optic neuropathy and platelet aggregation defect in Wolfram syndrome type 2.* BMC medical genetics, 2014. 15: p. 88.
174. Togashi, Y., et al., *Homozygous deletion of the activin A receptor, type IB gene is associated with an aggressive cancer phenotype in pancreatic cancer.* Molecular cancer, 2014. 13: p. 126.
175. Frey, U.H., et al., *PCR-amplification of GC-rich regions: 'slowdown PCR'.* Nature protocols, 2008. 3(8): p. 1312-7.
176. Phelan, K. and H.E. McDermid, *The 22q13.3 Deletion Syndrome (Phelan-McDermid Syndrome).* Molecular syndromology, 2012. 2(3-5): p. 186-201.
177. Kim, H.G. and L.C. Layman, *The role of CHD7 and the newly identified WDR11 gene in patients with idiopathic hypogonadotropic hypogonadism and Kallmann syndrome.* Molecular and cellular endocrinology, 2011. 346(1-2): p. 74-83.
178. Briggs, T.A., et al., *Tartrate-resistant acid phosphatase deficiency causes a bone dysplasia with autoimmunity and a type I interferon expression signature.* Nature genetics, 2011. 43(2): p. 127-31.
179. Vignoli, A., et al., *Medical care of adolescents and women with Rett syndrome: an Italian study.* American journal of medical genetics. Part A, 2012. 158A(1): p. 13-8.
180. *Large-scale discovery of novel genetic causes of developmental disorders.* Nature, 2015. 519(7542): p. 223-8.

181. Marshall Graves, J.A., *Human Y chromosome, sex determination, and spermatogenesis- a feminist view.* Biology of reproduction, 2000. 63(3): p. 667-76.
182. Falchi, M., et al., *Low copy number of the salivary amylase gene predisposes to obesity.* Nature genetics, 2014. 46(5): p. 492-7.
183. Lindstrand, A., et al., *Recurrent CNVs and SNVs at the NPHP1 Locus Contribute Pathogenic Alleles to Bardet-Biedl Syndrome.* American journal of human genetics, 2014. 94(5): p. 745-54.
184. Owen, J.P., et al., *Aberrant white matter microstructure in children with 16p11.2 deletions.* The Journal of neuroscience : the official journal of the Society for Neuroscience, 2014. 34(18): p. 6214-23.
185. van Duyvenvoorde, H.A., et al., *Copy number variants in patients with short stature.* European journal of human genetics : EJHG, 2014. 22(5): p. 602-9.
186. Cooper, G.M., et al., *A copy number variation morbidity map of developmental delay.* Nature genetics, 2011. 43(9): p. 838-46.
187. Cafferkey, M., et al., *Phenotypic features in patients with 15q11.2(BP1-BP2) deletion: Further delineation of an emerging syndrome.* American journal of medical genetics. Part A, 2014.
188. Preiksaitiene, E., et al., *Considering specific clinical features as evidence of pathogenic copy number variants.* Journal of applied genetics, 2014. 55(2): p. 189-96.
189. Qiao, Y., et al., *Variant ATRX syndrome with dysfunction of ATRX and MAGT1 genes.* Human mutation, 2014. 35(1): p. 58-62.
190. Fernandez-Rozadilla, C., et al., *A genome-wide association study on copy-number variation identifies a 11q11 loss as a candidate susceptibility variant for colorectal cancer.* Human genetics, 2014. 133(5): p. 525-34.
191. Dauber, A., et al., *SCRIB and PUF60 are primary drivers of the multisystemic phenotypes of the 8q24.3 copy-number variant.* American journal of human genetics, 2013. 93(5): p. 798-811.
192. Stefansson, H., et al., *Large recurrent microdeletions associated with schizophrenia.* Nature, 2008. 455(7210): p. 232-6.
193. Sanders, S.J., et al., *Multiple recurrent de novo CNVs, including duplications of the 7q11.23 Williams syndrome region, are strongly associated with autism.* Neuron, 2011. 70(5): p. 863-85.
194. Vulto-van Silfhout, A.T., et al., *Clinical significance of de novo and inherited copy-number variation.* Human mutation, 2013. 34(12): p. 1679-87.
195. Kashevarova, A.A., et al., *Array CGH analysis of a cohort of Russian patients with intellectual disability.* Gene, 2014. 536(1): p. 145-50.
196. Askree, S.H., et al., *Detection limit of intragenic deletions with targeted array comparative genomic hybridization.* BMC genetics, 2013. 14: p. 116.
197. Xia, X.Q., et al., *Evaluating oligonucleotide properties for DNA microarray probe design.* Nucleic acids research, 2010. 38(11): p. e121.
198. Caramaschi, E., et al., *Predictive diagnostic value for the clinical features accompanying intellectual disability in children with pathogenic copy number variations: a multivariate analysis.* Italian journal of pediatrics, 2014. 40(1): p. 39.
199. Ripke, S., et al., *Genome-wide association analysis identifies 13 new risk loci for schizophrenia.* Nature genetics, 2013. 45(10): p. 1150-9.

200. Schwab, S.G. and D.B. Wildenauer, *Genetics of psychiatric disorders in the GWAS era: an update on schizophrenia.* European archives of psychiatry and clinical neuroscience, 2013. 263 Suppl 2: p. S147-54.
201. Wiszniewska, J., et al., *Combined array CGH plus SNP genome analyses in a single assay for optimized clinical testing.* European journal of human genetics : EJHG, 2014. 22(1): p. 79-87.
202. Pinto, D., et al., *Comprehensive assessment of array-based platforms and calling algorithms for detection of copy number variants.* Nature biotechnology, 2011. 29(6): p. 512-20.
203. Kim, S.Y., J.H. Kim, and Y.J. Chung, *Effect of Combining Multiple CNV Defining Algorithms on the Reliability of CNV Calls from SNP Genotyping Data.* Genomics & informatics, 2012. 10(3): p. 194-9.
204. Sindi, S., et al., *A geometric approach for classification and comparison of structural variants.* Bioinformatics, 2009. 25(12): p. i222-30.
205. Castellani, C.A., et al., *Biological relevance of CNV calling methods using familial relatedness including monozygotic twins.* BMC bioinformatics, 2014. 15: p. 114.
206. Ehli, E.A., et al., *De novo and inherited CNVs in MZ twin pairs selected for discordance and concordance on Attention Problems.* European journal of human genetics : EJHG, 2012. 20(10): p. 1037-43.
207. Itsara, A., et al., *De novo rates and selection of large copy number variation.* Genome research, 2010. 20(11): p. 1469-81.
208. Wang, K., et al., *PennCNV: an integrated hidden Markov model designed for high-resolution copy number variation detection in whole-genome SNP genotyping data.* Genome research, 2007. 17(11): p. 1665-74.
209. Smith, B.H., et al., *Cohort Profile: Generation Scotland: Scottish Family Health Study (GS:SFHS). The study, its participants and their potential for genetic research on health and illness.* International journal of epidemiology, 2013. 42(3): p. 689-700.
210. Benaglia, T., et al., *mixtools: An R Package for Analyzing Finite Mixture Models.* Journal of Statistical Software, 2009. 32(6): p. 1-29.
211. Sharp, A.J., et al., *Segmental duplications and copy-number variation in the human genome.* American journal of human genetics, 2005. 77(1): p. 78-88.
212. Starke, H., et al., *Homologous sequences at human chromosome 9 bands p12 and q13-21.1 are involved in different patterns of pericentric rearrangements.* European journal of human genetics : EJHG, 2002. 10(12): p. 790-800.
213. Winchester, L. and J. Ragoussis, *Algorithm implementation for CNV discovery using Affymetrix and Illumina SNP array data.* Methods in molecular biology, 2012. 838: p. 291-310.
214. Lupski, J.R. and P. Stankiewicz, *Genomic disorders: molecular mechanisms for rearrangements and conveyed phenotypes.* PLoS genetics, 2005. 1(6): p. e49.
215. Belancio, V.P., P.L. Deininger, and A.M. Roy-Engel, *LINE dancing in the human genome: transposable elements and disease.* Genome medicine, 2009. 1(10): p. 97.
216. Koike, A., et al., *Comparative analysis of copy number variation detection methods and database construction.* BMC genetics, 2011. 12: p. 29.

217. Hehir-Kwa, J.Y., et al., *De novo copy number variants associated with intellectual disability have a paternal origin and age bias.* Journal of medical genetics, 2011. 48(11): p. 776-8.
218. Cooper, G.M. and H.C. Mefford, *Detection of copy number variation using SNP genotyping.* Methods in molecular biology, 2011. 767: p. 243-52.
219. Magi, A., et al., *EXCAVATOR: detecting copy number variants from whole-exome sequencing data.* Genome biology, 2013. 14(10): p. R120.
220. Bansal, V., *A statistical method for the detection of variants from next-generation resequencing of DNA pools.* Bioinformatics, 2010. 26(12): p. i318-24.
221. Koboldt, D.C., et al., *VarScan 2: somatic mutation and copy number alteration discovery in cancer by exome sequencing.* Genome research, 2012. 22(3): p. 568-76.
222. Gilissen, C., et al., *Genome sequencing identifies major causes of severe intellectual disability.* Nature, 2014. 511(7509): p. 344-7.
223. Krumm, N., et al., *Copy number variation detection and genotyping from exome sequence data.* Genome research, 2012. 22(8): p. 1525-32.
224. Campbell, C.D. and E.E. Eichler, *Properties and rates of germline mutations in humans.* Trends in genetics : TIG, 2013. 29(10): p. 575-84.
225. Crow, J.F., *The origins, patterns and implications of human spontaneous mutation.* Nature reviews. Genetics, 2000. 1(1): p. 40-7.
226. Lee, J.A., C.M. Carvalho, and J.R. Lupski, *A DNA replication mechanism for generating nonrecurrent rearrangements associated with genomic disorders.* Cell, 2007. 131(7): p. 1235-47.
227. Palmer, E., et al., *Changing interpretation of chromosomal microarray over time in a community cohort with intellectual disability.* American journal of medical genetics. Part A, 2014. 164A(2): p. 377-85.
228. Allen, A.S., et al., *De novo mutations in epileptic encephalopathies.* Nature, 2013. 501(7466): p. 217-21.
229. Knowles, M.R., et al., *Mutations in SPAG1 cause primary ciliary dyskinesia associated with defective outer and inner dynein arms.* American journal of human genetics, 2013. 93(4): p. 711-20.
230. Tanaka, M., et al., *Hyperglycosylation and reduced GABA currents of mutated GABRB3 polypeptide in remitting childhood absence epilepsy.* American journal of human genetics, 2008. 82(6): p. 1249-61.
231. Kyttala, M., et al., *MKS1, encoding a component of the flagellar apparatus basal body proteome, is mutated in Meckel syndrome.* Nature genetics, 2006. 38(2): p. 155-7.
232. Twigg, S.R., et al., *Reduced dosage of ERF causes complex craniosynostosis in humans and mice and links ERK1/2 signaling to regulation of osteogenesis.* Nature genetics, 2013. 45(3): p. 308-13.
233. Noor, A., et al., *Disruption at the PTCHD1 Locus on Xp22.11 in Autism spectrum disorder and intellectual disability.* Science translational medicine, 2010. 2(49): p. 49ra68.
234. Fiskerstrand, T., et al., *Familial diarrhea syndrome caused by an activating GUCY2C mutation.* The New England journal of medicine, 2012. 366(17): p. 1586-95.

235. Chung, B.H., V.Q. Tao, and W.W. Tso, *Copy number variation and autism: New insights and clinical implications.* Journal of the Formosan Medical Association = Taiwan yi zhi, 2014. 113(7): p. 400-408.
236. Harrison, S.M., et al., *DNA Copy-Number Variations in 46,XY Disorders of Sex Development.* The Journal of urology, 2014.
237. Gershon, E.S. and N. Alliey-Rodriguez, *New ethical issues for genetic counseling in common mental disorders.* The American journal of psychiatry, 2013. 170(9): p. 968-76.
238. Coughlin, C.R., 2nd, G.H. Scharer, and T.H. Shaikh, *Clinical impact of copy number variation analysis using high-resolution microarray technologies: advantages, limitations and concerns.* Genome medicine, 2012. 4(10): p. 80.
239. Riggs, E.R., et al., *Towards an evidence-based process for the clinical interpretation of copy number variation.* Clinical genetics, 2012. 81(5): p. 403-12.
240. de Ligt, J., et al., *Diagnostic exome sequencing in persons with severe intellectual disability.* The New England journal of medicine, 2012. 367(20): p. 1921-9.
241. O'Roak, B.J., et al., *Sporadic autism exomes reveal a highly interconnected protein network of de novo mutations.* Nature, 2012. 485(7397): p. 246-50.
242. Rauch, A., et al., *Range of genetic mutations associated with severe non-syndromic sporadic intellectual disability: an exome sequencing study.* Lancet, 2012. 380(9854): p. 1674-82.
243. Iossifov, I., et al., *De novo gene disruptions in children on the autistic spectrum.* Neuron, 2012. 74(2): p. 285-99.
244. Neale, B.M., et al., *Patterns and rates of exonic de novo mutations in autism spectrum disorders.* Nature, 2012. 485(7397): p. 242-5.
245. Szatkiewicz, J.P., et al., *Copy number variation in schizophrenia in Sweden.* Molecular psychiatry, 2014. 19(7): p. 762-73.
246. Rees, E., et al., *CNV analysis in a large schizophrenia sample implicates deletions at 16p12.1 and SLC1A1 and duplications at 1p36.33 and CGNL1.* Human molecular genetics, 2014. 23(6): p. 1669-76.
247. Krumm, N., et al., *Transmission disequilibrium of small CNVs in simplex autism.* American journal of human genetics, 2013. 93(4): p. 595-606.
248. Carvalho, C.M., et al., *Evidence for disease penetrance relating to CNV size: Pelizaeus-Merzbacher disease and manifesting carriers with a familial 11 Mb duplication at Xq22.* Clinical genetics, 2012. 81(6): p. 532-41.
249. Kirkpatrick, R.M., et al., *Low-Frequency Copy-Number Variants and General Cognitive Ability: No Evidence of Association.* Intelligence, 2014. 42: p. 98-106.
250. Smith, B.H., et al., *Generation Scotland: the Scottish Family Health Study; a new resource for researching genes and heritability.* BMC medical genetics, 2006. 7: p. 74.
251. Szatmari, P., et al., *Mapping autism risk loci using genetic linkage and chromosomal rearrangements.* Nature genetics, 2007. 39(3): p. 319-28.
252. Ingason, A., et al., *Copy number variations of chromosome 16p13.1 region associated with schizophrenia.* Molecular psychiatry, 2011. 16(1): p. 17-25.

253. Hoischen, A., et al., *De novo mutations of SETBP1 cause Schinzel-Giedion syndrome.* Nature genetics, 2010. 42(6): p. 483-5.
254. Minakuchi, M., et al., *Identification and characterization of SEB, a novel protein that binds to the acute undifferentiated leukemia-associated protein SET.* European journal of biochemistry / FEBS, 2001. 268(5): p. 1340-51.
255. Dabell, M.P., et al., *Investigation of NRXN1 deletions: clinical and molecular characterization.* American journal of medical genetics. Part A, 2013. 161A(4): p. 717-31.
256. Harrison, V., et al., *Compound heterozygous deletion of NRXN1 causing severe developmental delay with early onset epilepsy in two sisters.* American journal of medical genetics. Part A, 2011. 155A(11): p. 2826-31.
257. Chen, Q., et al., *Homozygous deletion in KVLQT1 associated with Jervell and Lange-Nielsen syndrome.* Circulation, 1999. 99(10): p. 1344-7.
258. Jongbloed, R.J., et al., *Novel KCNQ1 and HERG missense mutations in Dutch long-QT families.* Human mutation, 1999. 13(4): p. 301-10.
259. Dolcetti, A., et al., *1q21.1 Microduplication expression in adults.* Genetics in medicine : official journal of the American College of Medical Genetics, 2013. 15(4): p. 282-9.
260. Grozeva, D., et al., *Independent estimation of the frequency of rare CNVs in the UK population confirms their role in schizophrenia.* Schizophrenia research, 2012. 135(1-3): p. 1-7.
261. Magri, C., et al., *New copy number variations in schizophrenia.* PloS one, 2010. 5(10): p. e13422.
262. Elias, P.M., et al., *Role of cholesterol sulfate in epidermal structure and function: lessons from X-linked ichthyosis.* Biochimica et biophysica acta, 2014. 1841(3): p. 353-61.
263. Murtaza, G., et al., *Molecular study of X-linked ichthyosis: report of a novel 2-bp insertion mutation in the STS and a very rare case of homozygous female patient.* Journal of dermatological science, 2014. 74(2): p. 165-7.
264. Sever, R.J., P. Frost, and G. Weinstein, *Eye changes in ichthyosis.* JAMA : the journal of the American Medical Association, 1968. 206(10): p. 2283-6.
265. Went, L.N., et al., *X-linked ichthyosis: linkage relationship with the Xg blood groups and other studies in a large Dutch kindred.* Annals of human genetics, 1969. 32(4): p. 333-45.
266. Bradshaw, K.D. and B.R. Carr, *Placental sulfatase deficiency: maternal and fetal expression of steroid sulfatase deficiency and X-linked ichthyosis.* Obstetrical & gynecological survey, 1986. 41(7): p. 401-13.

Publications

Publications arising from work associated with this thesis:

- **Fitzgerald TW**, Larcombe LD, Le Scouarnec S, Clayton S, Rajan D, Carter NP, Redon R (2011). "aCGH.Spline--an R package for aCGH dye bias normalization." Bioinformatics (9):1195-200.

- Redon R, **Fitzgerald T**, Carter NP (2009). "Comparative genomic hybridization: DNA labelling, hybridization and detection." Methods Mol Biol. 2009;529:267-78.

- **Fitzgerald, T. W**., S. S. Gerety, et al. (2015). "Large-scale discovery of novel genetic causes of developmental disorders." **Nature** **519**(7542): 223-228

- Wei W., **T. W. Fitzgerald**, et al. (2015). "Copy number variation on the human Y chromosome in UK control males from the Deciphering Developmental Disorders Project" (**Human Genetics**)

- Wright, C. F., **T. W. Fitzgerald**, et al. (2014). "Genetic diagnosis of developmental disorders in the DDD study: a scalable analysis of genome-wide research data." **Lancet**

- King, D. A., **T. W. Fitzgerald**, et al. (2014). "A novel method for detecting uniparental disomy from trio genotypes identifies a significant excess in children with developmental disorders." **Genome research** **24**(4): 673-687

Manuscripts under revision

- **Chapter 4: Fitzgerald TW**, Morley KI,, King DA, van Kogelenberg M, Sifrim A, Jones W, Clayton S, Jones P, Parthiban V, Bragin E, Tivey AR, Gribble S, Prigmore E, Rajan D, Ambridge K, Barrett D, Krishnappa N, Bayzetinova T, Jostins L, Dominiczak A, Morris A, Porteous D, Smith B, Mohareb F, Carter NP, FitzPatrick DR, Firth HV, Wright CF, Barrett JC and Hurles ME, on behalf of the DDD study "Improved CNV discovery algorithms enable an exon-level resolution map of CNV and reappraisal of the CNV mutation rate" (**Genome Research)**

Manuscripts in preparation

- **Chapter 5: Fitzgerald TW**, Gribble S, Prigmore E, Rajan D, Ambridge K, Barrett D, Krishnappa N, Bayzetinova T, Mohareb F, Carter NP, FitzPatrick DR, Firth HV,, Wright CF, Barrett JC and Hurles ME, on behalf of the DDD study "Potentially Clinically Relevant Exon Resolution Copy Number Variation in Normal Control Samples from the Deciphering Developmental Disorders Project." (**Genome Research**)

7 | Appendix

7.1 CNsolidate Parameters

7.1.1 Simulation Methods

In CNsolidate there are a number of simulation-based methods that can help to optimize certain necessary parameters for data of different types. These methods are capable of learning certain characteristics of the input data under consideration.

Given a series of training data sets the simulation methods will calculate the central and difference based noise measures as well as the overall range and scale of the data values. These measures are saved as R objects and used during the generation of synthetic data sets. Additionally a custom wavelet transform algorithm is used to extract a number of different frequency components ('octaves') from the training data sets. These are again saved as objects and used to add varying degrees of auto-correlation ('wave') effects to the synthetically generated data sets. Furthermore, a method that adds a given number of randomly dispersed single outlier data points across the range is included. This is aimed at making the generated synthetic data characteristic more realistic. The range of the number of single outlier data points to be added is again estimated from the input training datasets.

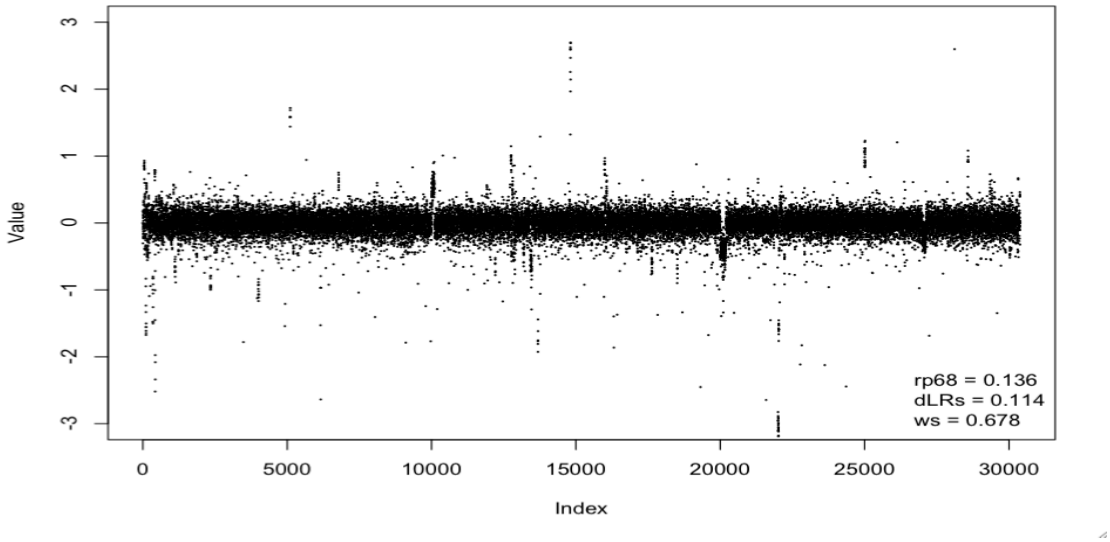


Figure 7-7-1 Example of a synthetically generated array-CGH data set

Above (see **Figure 7-1**) is an example of a synthetically generated data set from the synthetic test set. The parameter ranges used were estimated using a training set made up of close to 300 individual real array-CGH data sets. To create the set of synthetic test data sets, the mentioned parameters were randomized 1 million times, generating a unique synthetic data set for each set of parameters. Each data set is made up of 24 subsets (relating to the chromosomes of the input training data arrays), each of the subsets contains 10000 data points, giving a total of 240000 data point per synthetic data set. The above plot shows

'chromosome1' from a particular synthetic data set, which displays relatively low noise values for the three data estimators, rp68, dLRs and ws. The 1 million synthetic test data sets contain a large number of different combinations of noise characteristics. Below is an example showing the effect of adding a relatively high amplitude wave component from the frequency component bank (derived from the training data sets).

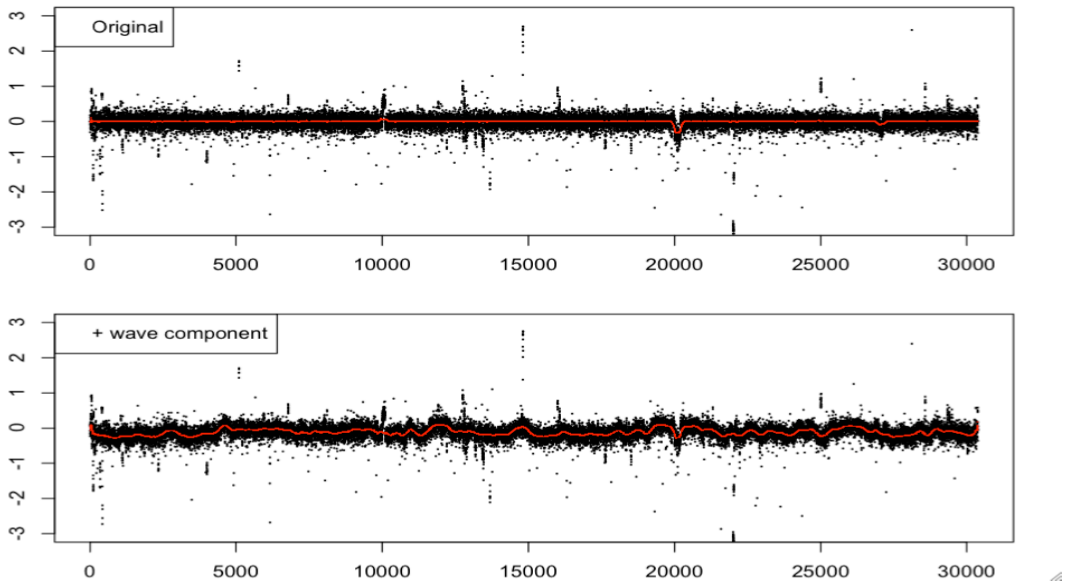


Figure 7-7-2 Example of adding a wave component

On the top panel (see **Figure 7-2**) chromosome1 from the original synthetic data set and on the lower panel chromosome1 after adding a training data derived wave component is shown. The ws value in the top panel is 0.678, whereas in the bottom panel, after adding the wave component, the ws has increase to a value of 2.38. The rp68 value for the top panel is 0.136 whereas, for the bottom panel it is 0.212. The dLRs value for the top panel is 0.11458 whereas, for the bottom panel it is 0.11450 (slightly decreased). This provides a reasonable illustration for the difference between the three included noise estimators and why it is important to use multiple estimates of data noise characteristics.

The rp68, being a measure of the scale of noise within the centralized value region, provides a measure of the normally distributed noise within the 68th percentile of the median normalized value distribution. The dLRs, being based on the IQR between the differences between consecutive data points, provides an approximation of the difference between consecutive values needed to define an outlier, assuming that the data is normally distributed. The ws, being designed to search for auto-correlation across a time-series, provides a reasonable approximation of the level at which the data is actually 'normally' distributed, assuming auto-correlation is the only difference to normality.

The main benefit of using simulation to estimate algorithm performance is that it is possible to obtain a reliable measure of the false and true positive rates. The reason for this is that during the data simulation synthetic CNV events are

inserted and their location and characteristics are recorded. Therefore by comparing change point detection results against the pre-known CNV locations it is possible to assess how successful individual algorithm were at **1)** detecting the inserted change point interval (CNVs) – the TP rate and **2)** How many erroneous detections (those not pre known) were made – the FP rate.

On top of being able to generate reliable TP and FP rates for individual algorithms overall, by using this type of simulation approach it is possible to generate different rates given various predictive variables and scales. As mentioned above three different data noise measures were varied across the simulations allowing data type and level specific FP and TP rate estimations. Furthermore, during the insertion of synthetic CNV events it was possible to vary some additional parameters relating to specific characteristics of the CNV events themselves. For this the mean log2 ratio, the number of data points and the variance across data points within a synthetic CNV were used as the predictive variables. It was then possible to estimate the effect that differences in the values of these 3 variables had on the FP and TP rates overall (the feature dependant weighting functions).

6.1.2 Optimized Algorithm Parameters

One complication with using such a large number of algorithms in combination is that each algorithm has a number of different parameters that need to be optimized. This can be a time consuming and complex task even for a single algorithm. However, when considering multiple algorithms, the number of possible combinations for parameter definitions quickly becomes large. The simulation-based methods included in CNsolidate allow this process to be semi-automated. They do not only work in the context of single algorithms, the optimization of parameters can be assessed in the context of the combined performance of all algorithms. This means that the optimized set of parameter definitions are estimated based upon on the performance of all algorithms in concert. During this parameter optimization process, all algorithms are assumed to contribute equal weight to individual change point detections.

6.1.3 Approximating Weighting Functions

Individual algorithm performances vary given differing data characteristics. We approximate this behaviour by deriving specific weighting functions for each algorithm given certain data characteristics.

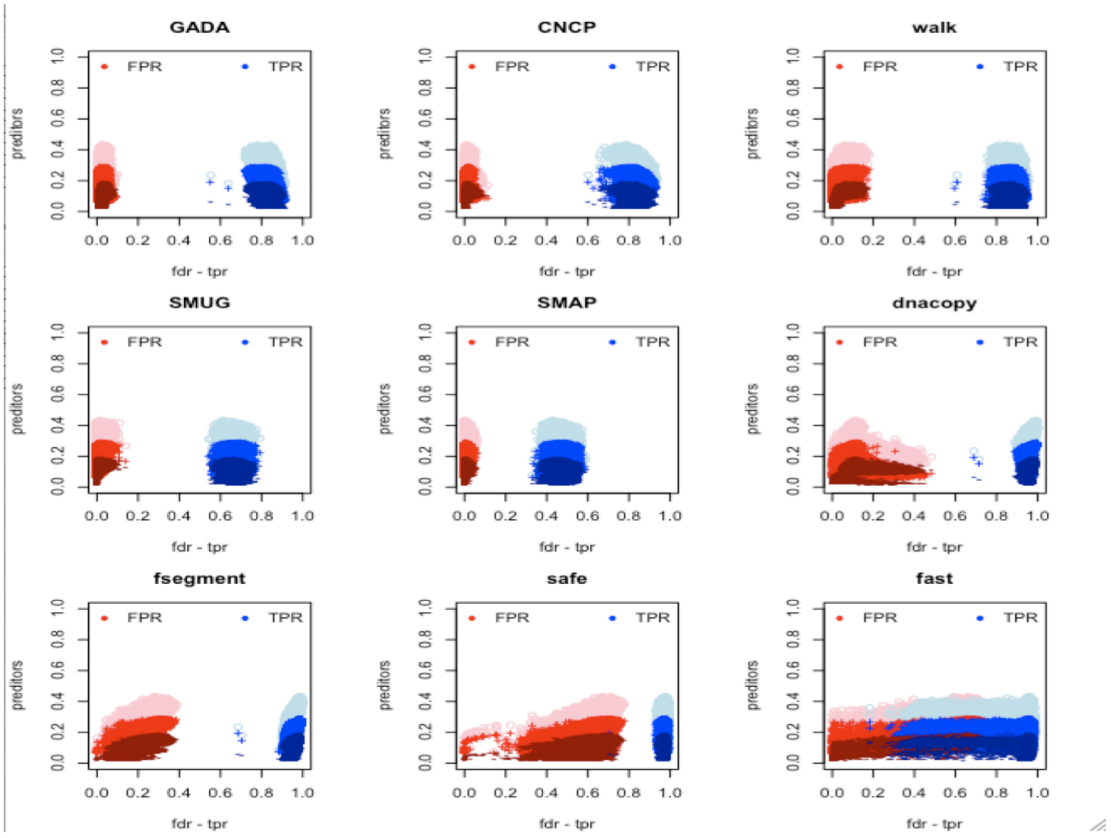


Figure 7-7-3 Example of deriving weighting functions based on estimated individual algorithm performances

Above (see **Figure 7-3**) is an example of the results of assessing the performance of individual algorithms using data simulation in terms of the estimated false positive (FP) and true positive (TP) rates given the noise type and range of the synthetic data sets. The 9 panels displayed show both the FP (red) and TP (blue) rates against a measure of noise across over 1 million data simulations. The shade of the data points on the plots relate to the different noise predictors used.

6.2 Implementation

All detection and combination methods described above are contained inside a single R package with no external dependencies. It is completely platform independent and can be installed on any machine with a valid R installation. The package also contains some normalization methods appropriate for array-CGH data. There are splines and lowess based dye-bias normalization methods, a general regression based method and a custom wavelet transform. The code base contains approximately, 138 R scripts, 16 Java classes, 16 C classes, 4 C++ classes and a few Fortran methods (see package documentation for full function definition and usage).

All parameter and weighting function definitions have some default options that can be used and should perform well across a range of data types. However there are also some simulation based methods included which allow the semi-automated definition of individual algorithm parameters and weighting functions.

These simulation-based methods are capable of learning certain characteristics of the input data type. The methods can perform large (or small) scale simulations to allow some necessary parameters to be accurately estimated for the specific data type. The results of these simulation methods are R objects that can be saved and referenced during subsequent analysis.

6.2 DDD Validation Array Design

6.2.1 Objective

The primary aim of this experiment is to select a set of CNVs for validation to permit estimation of the false discovery rate for the DDD CNV discovery pipeline, and fine-tuning of the algorithms used to detect CNVs. This experiment is not designed to evaluate estimation of *de novo* CNV status, so only proband samples will be included on the array, but these will be selected from complete trios.

6.2.2 Sample Selection

There is a trade-off between minimising the number of samples required for the validation experiment, and making sure enough sample DNA remain for the DDD pipelines and other future experiments. As the aim is to validate both the array-CGH and Exome CNV calling algorithms, samples will be selected from the pilot trios.

Forty trios meeting the following criteria were identified:

1. The proband has at least one high-quality set of array-CGH and Exome calls
2. There is sufficient (> 1 ug) DNA for all members of the trio

A number of the probands have two sets of array-CGH data, one from blood DNA and another from salivary DNA. In each instance the sample that generated the highest quality call-set will be used for the validation experiment. Where there is no difference in call quality, the sample with the largest amount of DNA available will be used.

6.2.4 CNV selection strategy

This validation design will try to validate calls from both the array-CGH and Exome platform so CNVs that are called in only one dataset, and those that are called in both need to be considered. CNVs will be selected from the set of pilot patients with high-quality data from both sources.

1. High quality call sets for the 32 samples are generated for both array-CGH and Exome platforms
2. Each of these is split into common and rare categories using the CNV consensus v2.1 as the reference set
3. To define common, a > 0.8 forward overlap and > 0.5 backward overlap for array-CGH and > 0.8 forward overlap and > 0 backward overlap for Exome, from CNVs with a 1% or greater population frequency.
4. The proportion of common to rare CNVs to select is set at 25% and 75% respectively.

Using the CGH 8 x 60 format select 36 CNV regions per sample (9 common and 27 rare). This strategy is independent of detection quality scores in both cases, however the call lists used for selection are 'pre-filtered' based on detection quality scores. The goal is to include a relatively small number of edge cases near the currently defined quality score cut-offs.

After selection it is acceptable to generate some summary statistics about the distribution of various characteristics of the selected CNV regions, combined with the visual inspection of a number of events. This should be used to check that the selected CNV set is generally sensible but not used to exclude individual CNV regions. If the final selected CNV validation set displays general poor characteristics the random selection procedure must to performed again.

6.2.5 Analysis

Visual inspection can be used to confirm or investigate findings, but it is not feasible to do this for all CNVs included in the experiment. Statistical analysis of the data could take a variety of approaches. Individual summary statistics for the intensity at CNV can be compared either to all other regions within an individual, or across all other samples at the same region. The intensity of probes within the CNV break points can be compared to those outside (within an individual), although this assumes that the break points have been accurately captured in the original experiment.

6.2.6 Timeline

Table 7-1 Validation array processing timeline

Step	Time	Details
Array design	3 weeks	Out-sourced to Agilent
Array production	6 weeks	Upper limit of estimated delivery time
Scale-down of array processing	2 weeks	Partial overlap with production time
Array processing	2 weeks	Can process 100 slides per fortnight
Analysis	3 weeks	
Total	16 weeks	

6.2.7 Data Quality Measures – array-CGH and Exome

The 32 samples are selected based on data quality measures from both the array-CGH and Exome platforms. For the array-CGH quality measure the mean dLRs (derivative log2 ratio spread) between both arrays (arrayA and arrayB) is used, whereas for the Exome platform use the MAD (median absolute deviation) of the log2 ratio values is used (see **Figure 7-4**).

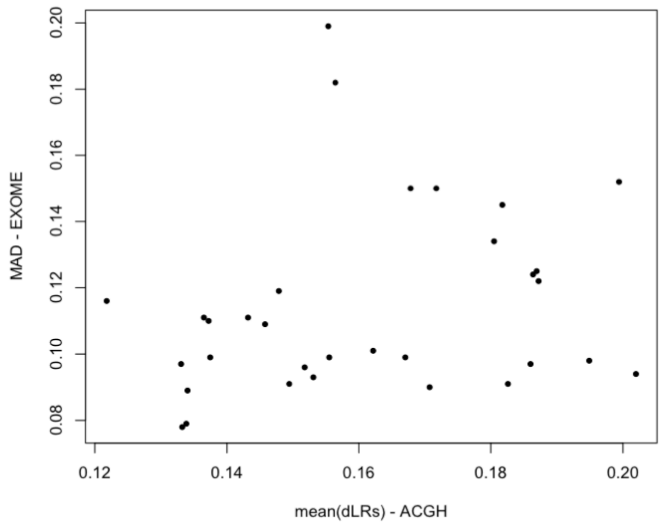


Figure 7-7-4 Noise measures in array-CGH against Exome across 32 validation samples

Above (see **Figure 7-4**) is a plot of the quality measures from array-CGH and Exome platforms for each of the 32 selected samples. Overall the data quality measures are relatively low, denoting high quality data from both platforms. Furthermore, the data quality from both platform displays a reasonable range of values across the high quality band (0.05 - 0.2 Exome and 0.1 - 0.2 array-CGH).

6.2.8 Data Quality Measures – array-CGH

Additionally, for the array-CGH platform, we look at the agreement of three different QC measures between the two arrays for each sample (arrayA and arrayB).

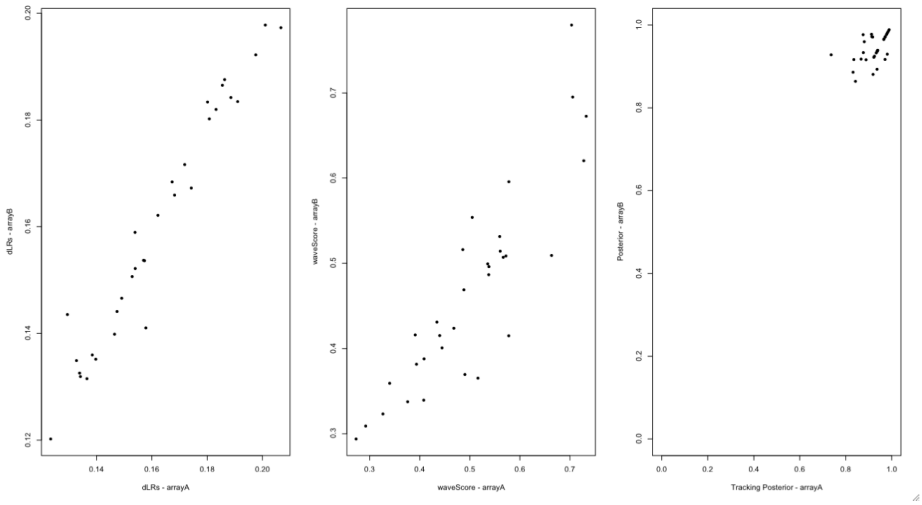


Figure 7-7-5 Noise measure between arrayA and arrayB from the array-CGH platform across 32 validation samples Left - dLRs, Middle - wave score and Right - tracking posterior

Above (see **Figure 7-5**) are three plots showing the agreement, for each sample, between arrayA and arrayB on the array-CGH platform. Generally, the agreement is reasonable for each validation sample across all QC values.

6.2.9 Array Design

The 8 x 60 Agilent format is most sensible for this validation experiment (see **Table 7-1**). This format, using 32 samples, can yield a total of 2304 CNVs regions (72 per sample) for validation. This has to be split across the two platforms, meaning that 36 CNV regions can be selected from each sample for both the array-CGH and Exome platforms.

Additional probes - Control Probes:

1. 10000 random probes selected from the DDD array-CGH design that are not called in any of the 32 individuals - evenly spread between the 22 autosomes
2. 300 CNV tracking probes - same as on the DDD array-CGH design.
3. 30 gender markers - same as on the DDD array-CGH design.

After the inclusion of the additional probes there are a total of 48760 probes for CNV region validation. Therefore, for each CNV region, for validation there are approximately 21 probes available (48760/2304 = 21.16).

6.2.10 CNV Region Selection - Call Sets

High quality calls sets from both array-CGH and Exome detection methods are supplied for the 32 samples. CNsolidate with default parameters is used for CNV detection from the array-CGH data. Detection quality measures 'wscore' and 'p value' are used to pre-filter the detections, with a cut-off of >= 0.5 and <= 0.001 respectively. CoNVeX using t=5 and p=2 is used for CNV detection from the Exome data, the 'CoNVex score' is used to prefilter the detections using a cut-off of 5 (t=5).

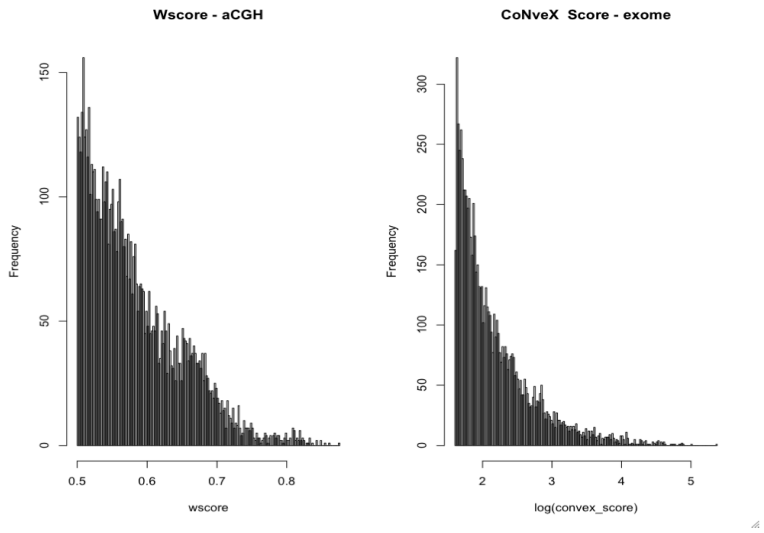


Figure 7-7-6 Detection quality scores for the selectable CNVs across the 32 samples for array-CGH (left) and Exome (right)

The two distributions above (see **Figure 7-6**) are somewhat similar however the log of the 'convexscore' is more truncated and has a larger range. Whereas the 'wscore' has a range between 0 - 1 and seems to displays a slightly greater proportion of quality scores towards the higher end.

Table 7-2 Summary of the overall number of CNV detections made from array-CGH and Exome across the 32 validation samples

	Array-CGH	Exome
Min	146	99
1st Qu.	197	207
Median	220	230
Mean	216	241
3rd Qu.	232	261
Max	290	444

The overall number of detections for the 32 samples is 6929 and 7733 for array-CGH and Exome respectively (see **Table 7-3**). Above is a table showing a summary of the number of detections for each sample from array-CGH and Exome respectively. All of these CNV calls are available for the selection procedure.

6.2.11 array-CGH and Exome CNV Selection

The high quality detection lists for the 32 samples are used to select CNV regions for validation using the previously described approach.

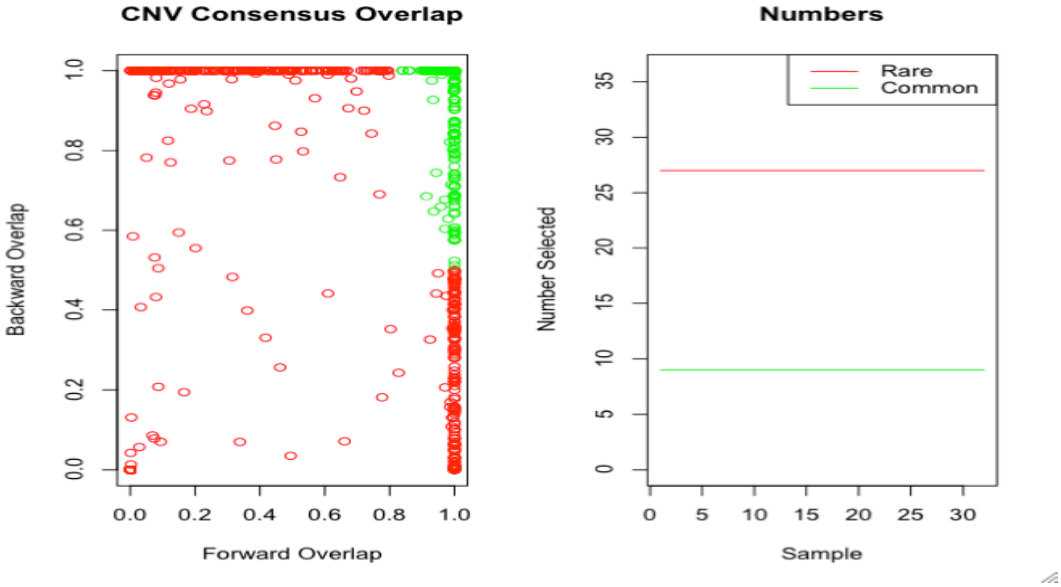


Figure 7-7 The overlap with the CNV Consensus2.1 of the selected validation CNVs (left) and the number of rare and common selected validation CNVs per sample (right) from the array-CGH platform

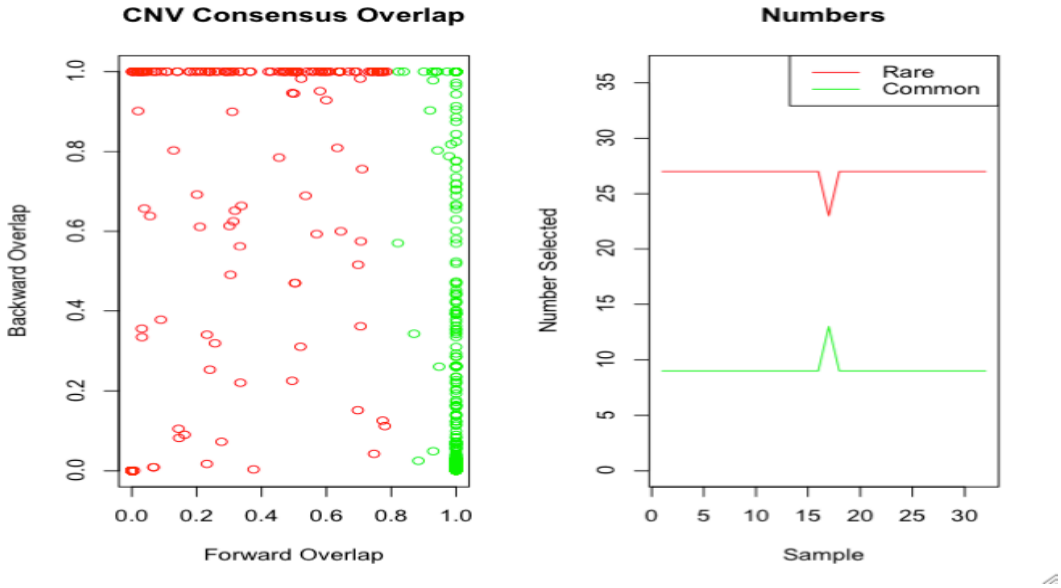


Figure 7-8 The overlap with the CNV Consensus2.1 of the selected validation CNVs (left) and the number of rare and common selected validation CNVs per sample (right) from the Exome platform

Above is a plot of overlap against the CNV consensus2.1 and the total number of selected CNV region in the common and rare categories per sample (see **Figure 7-7** & **Figure 7-8**). This is reassuring since **1)** the classification of rare and common is accurate given the defined parameters and **2)** the total number of selected CNVs in each category per sample is close to that defined (9 common and 27 rare per sample).

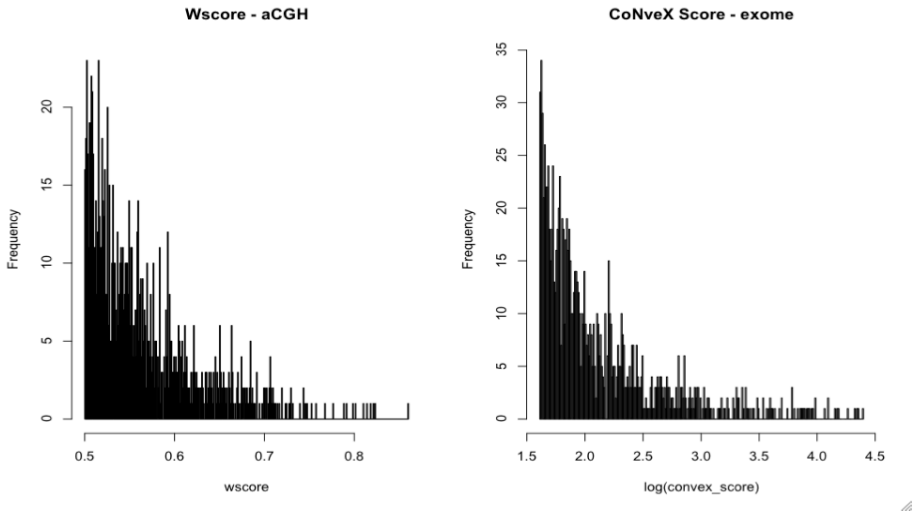


Figure 7-7-9 Detection quality scores for the selected CNVs across the 32 samples for array-CGH (left) and Exome (right)

Above (see **Figure 7-9**) are histograms of the detection quality measures for the selected CNV regions. In both cases this is a fair reflection of the distribution of detection quality measures of the 'selectable' CNV calls sets (see **Figure 7-6**).

The selection procedure is totally random within the rare and common categories. Below is a summary of the numbers of rare and common variants selected for the array-CGH and Exome platforms.

Table 7-3 Numbers and types of selected validation CNVs

	Total	Rare	Common	p.common	p.nonredundant
Array-CGH	1152	864	288	0.26	0.87
Exome	1152	860	292	0.25	0.84

The number of redundant CNV regions selected (exactly the same break point positions) is 13% and 16% for array-CGH and Exome respectively (see **Table 7-4**). Furthermore, there are 574/1152 (49%) and 683/1152 (59%) selected CNV regions that display an overlap (any overlap) with at least 1 other selected CNV region, for array-CGH and Exome data respectively.

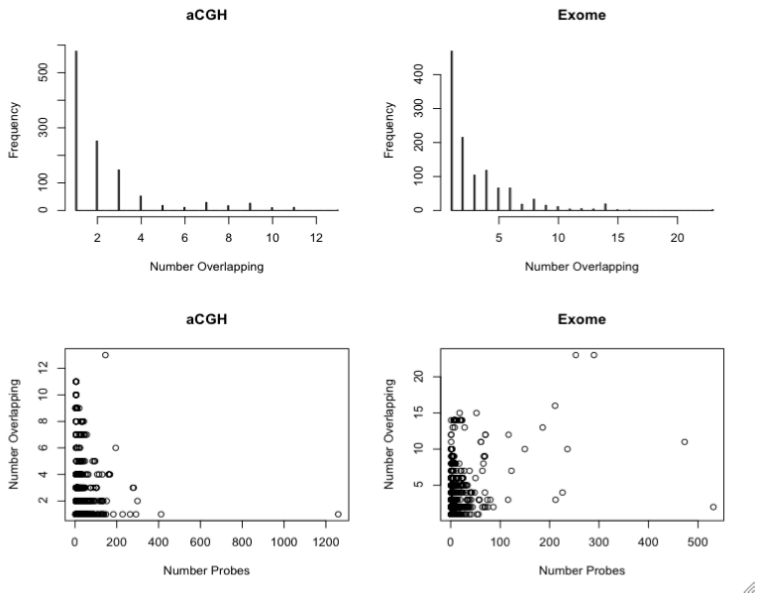


Figure 7-7-10 Overlapping CNVRs from the selected validation CNVs in array-CGH (left) and Exome (right)

Above (see **Figure 7-10**), for each of the selected CNV regions, for each set individually (array-CGH and Exome), the number of other CNV regions, from the set, that show an overlap (any overlap) with the selected region is shown.

Most of the selected regions are unique to that set, 87% and 84% for array-CGH and Exome respectively. Those remaining display exactly the same break points and cannot be considered as specific events. The larger number of overlaps in the exome data is likely due to smaller (exon) level events detected within a larger called region. These are highly interesting for inclusion and may help to optimise any merging issues. These comparisons are using the selected CNV regions, not the full calls lists.

6.2.12 array-CGH Selected vs. Exome Selected CNV Regions

The previous section compares each set individually; here each set (array-CGH and Exome) is compared against one another. Overall, there are 225/1152 (19.5%) and 280/1152 (24.3%) CNV regions that display any overlap with the other set, for the array-CGH and Exome sets respectively.

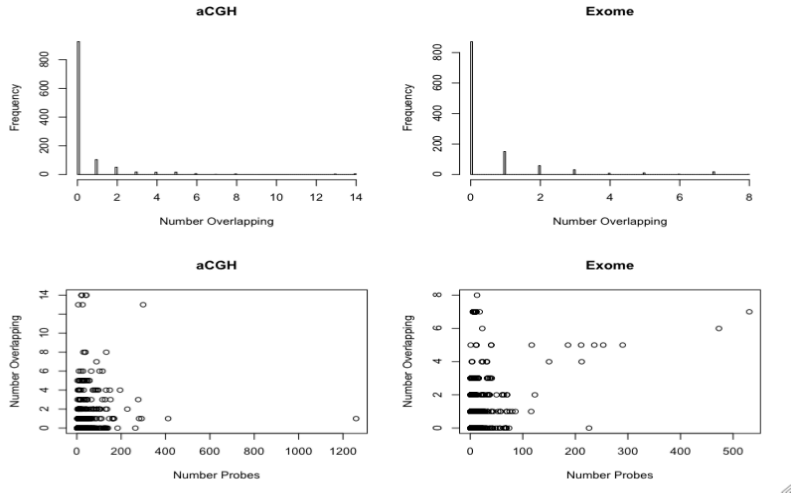


Figure 7-7-11 Overlapping CNVRs from the selected validation CNVs between array-CGH (left) and Exome (right)

Above (see **Figure 7-11**) are some plots to display the overall distribution of the overlaps between sets. Per sample, for any overlap of selected CNV regions between the sets (array-CGH and Exome) the total number is 52/1152 (4.5%) and the median is 1.5 per sample.

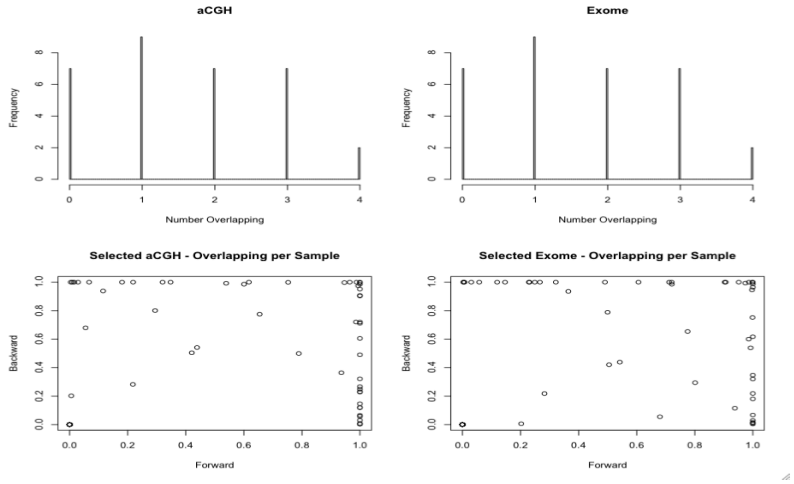


Figure 7-7-12 Overlapping CNVRs CNVRs from the selected validation CNVs between array-CGH (left) and Exome (right) per sample

There are 7 samples that have no selected CNV region that overlaps between the sets (see **Figure 7-12**). However, the number of overlapping regions per sample is not dominated by samples showing high numbers of overlaps (which could be

due e.g. an over-segmentation of calls). The majority of samples (25) show a small number (1-4) of selected CNV regions that are shared between sets per sample.

6.2.13 Summary

Overall the current selected CNV calls are a reasonable representation of the all CNV calls made across the 32 samples. The selection method is available as a single R script with a dependency for an R package. The selection can be re-run from start the finish very easily and the various parameters can be modified using the 'selection.R' script. See the README file for instructions. This will result in a number of plots, summaries and output files.

6.2.14 Validation - Data QC

Array results (validation array) were obtained for the 32 samples run on the 105K custom Agilent array from the DDD high-throughput array-CGH laboratory. The 8 x 105K slides were scanned using an Agilent scanner with default settings and the linear dye bias normalisation option.

After normalisation the 32 data sets were interrogated for data quality using a custom data tracking measure. This measure is based on the concordance between approx. 30 CNV tagging SNPs typed on a Sequenom assay and the estimated copy number states at the associated CNVR (see **Chapter 2**).

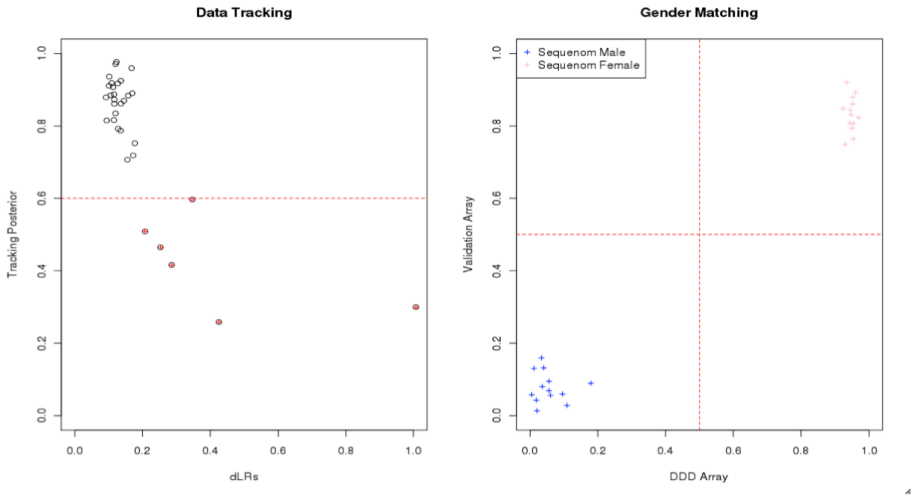


Figure 7-7-13 Data Tracking QC for the validation array, Left - the data tracking posterior vs. a noise measure (dLRs) for the 32 validation data sets, Right - the estimated gender of the 32 samples type on the discovery array (array-CGH) vs. the gender type on the validation array

As shown on the left panel of the above plot (see **Figure 7-13**), a greater than 0.6 cut-off on the tracking posterior is used to define samples that pass data QC. Using this cut-off resulted in failing 6 / 32 samples. All passed data sets display a dLRs of less than 0.2 and there were no gender mismatches between the discovery and validation array overall as indicated in the right panel (see **Figure 7-13)**. By using the custom data tracking approach the DNA samples typed on

the validation array are predicted to be the same as those typed on the discovery array.

6.2.15 Validation - Array Design QC

To check the accuracy of probe placement of the validation array two statistics, 'Edge', and 'Spread' were used. 'Edge' is the proportion of probes in a CNV that are within the first and last 10% of the CNV. 'Spread' is the distance from the start of the first probe to the end of the last probe within a CNV, scaled to the length of the CNV (i.e. the maximum spread is 1).

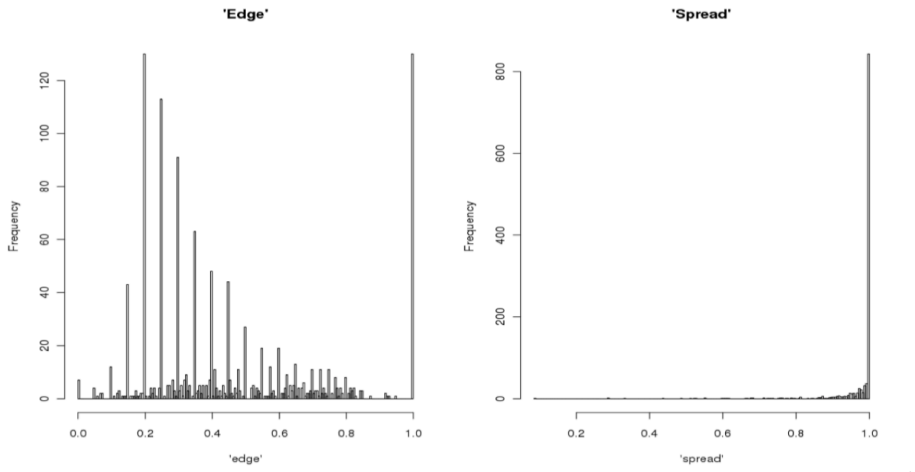


Figure 7-7-14 Two statistics summarizing probe placement within the selected validation CNVs, Left - the 'Edge', or proportion of probes in the first and last 10% of the CNV, Right - 'Spread' the distance between the first and last probe within the CNV, scaled by the length of the CNV.

The two plots above (see **Figure 7-14**) show that the probe placement within the targeted CNVR is reasonable. Observe that, indicated by the 'Spread' measure; the majority of CNVRs are fully covered ('Spread' value tending towards 1). Additionally, the 'Edge' measure shows that very few CNVRs have no probes in the first and last 10% of the CNVR ('Edge' == 0). Furthermore the majority of 'Edge' values are between a value of 0-1 and approximately centred between 0.2-0.3, meaning that the probe placement in the majority of CNVRs in even across the region.